IBM

IBM Enterprise Content Management and System Storage Solutions: Working Together

Business drivers for ECM

Matches storage products to ECM requirements

Provides practical case studies

Mary Lovelace
Nicholas Buchanan
Gavin Cameron
Flavio de Rezende
John Tarella

Redbooks

ibm.com/redbooks

International Technical Support Organization

IBM Enterprise Content Management and System Storage Solutions: Working Together

September 2008

SG24-7558-00

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

First Edition (September 2008)

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at `http://www.ibm.com/legal/copytrade.shtml`

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®
DB2®
DB2 Universal Database™
DFSMS™
Domino®
DS4000™
DS6000™
DS8000™
ESCON®
FICON®
FileNet®
FlashCopy®
General Parallel File System™
GPFS™
HACMP™
IBM®
Lotus®
Nextra™
OS/400®
POWER™
POWER5+™
Redbooks®
Redbooks (logo) ®
System i™
System p™
System Storage™
System Storage DS™
System x™
System z™
Tivoli®
TotalStorage®
Virtualization Engine™
WebSphere®
Workplace™
XIV®
z/OS®
zSeries®

The following terms are trademarks of other companies:

Acrobat, and Portable Document Format (PDF) are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

FileNet, and the FileNet logo are registered trademarks of FileNet Corporation in the United States, other countries or both.

Snapshot, LockVault, SyncMirror, SnapVault, SnapRestore, SnapMirror, SnapManager, SnapLock, SnapDrive, MultiStore, FlexVol, FlexClone, FilerView, Data ONTAP, and the NetApp logo are trademarks or registered trademarks of NetApp, Inc. in the U.S. and other countries.

Oracle, JD Edwards, PeopleSoft, Siebel, and TopLink are registered trademarks of Oracle Corporation and/or its affiliates.

SAP, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

EJB, J2EE, Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

An Enterprise Content Management (ECM) system is designed to contain unstructured information such as files, images, and drawings. Its purpose is the delivery of the right content to the right person at the right tme, and in the right context.

In enterprise content management, the term *content* is used to refer to unstructured information; while structured information such as database content is referred to as *data*. Although *data* is also present in ECM systems, it is used in a supportive role to help locate and manage the *content*.

This IBM® Redbooks® publication will provide the necessary information to IBMers, business partners, and customers on how to implement FileNet® ECM with IBM Storage Solutions.

The team that wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, San Jose Center.

Mary Lovelace is a Consulting IT Specialist at the International Technical Support Organization. She has more than 20 years of experience with IBM in large systems, storage and storage networking product education, system engineering and consultancy, and systems support. She has written many IBM Redbooks on z/OS storage products and TotalStorage Productivity Center.

Nicholas Buchanan is a Managing Consultant with IBM ECM Lab Services for IBM US. He has over twelve years of experience developing large scale information delivery solutions and brings extensive application development experience to IBM. Nicholas currently consults on IBM FileNet ECM architecture and solution design and has written extensively on high availability and business continuity.

Gavin Cameron is a Technical Specialist with IBM Software Group Lab Services for IBM Australia and has worked for IBM for the past 3 years. Prior to working for IBM he was with an IBM Business Partner. Gavin has over seven years of experience in the architecture and implementation of large scale IBM Enterprise Content Management solutions. He holds a degree in Computer Systems Engineering from the Royal Melbourne Institute of Technology in Melbourne, Australia.

Flavio de Rezende is a Software IT Architect in the Americas GTS SO SW Sales team. He has over four years of experience design and implementation of large scale IBM Enterprise Content Management solutions. Flavio also has extensive experience as a DBA and system administrator for content management systems. He holds a Ph.D. in Environmental Engineering from Penn State University.

John Tarella is a Senior Consulting IT Specialist who works for IBM Global Services in Italy. He has twenty years of experience in storage and performance management on mainframe and distributed environments. He holds a degree in Seismic Structural Engineering from Politecnico di Milano, Italy. His areas of expertise include IBM Tivoli® Storage Manager and SAN consulting, design, implementation services, and open systems storage, and storage performance monitoring and tuning. He is presently focusing on storage solutions for business continuity, information lifecycle management, and infrastructure simplification. He has written extensively on z/OS® DFSMS™, IBM Tivoli Storage Manager, SANs, storage business continuity solutions, and ILM.

Thanks to the following people for their contributions to this project:

Charlotte Brooks
International Technical Support Organization, San Jose Center

Michael Barton
Offering Manager, Enterprise Content Management (ECM) Solutions

Jerry Bower
Offering Manager, Storage Integrations, IBM Enterprise Content Management

Richard Heffel
Software Performance Analyst with the IBM Software Group's Content Manager development lab in Santa Teresa.

Holger Koenig
Content Management Center of Excellence in Boeblingen, Germany.

David Lebutsch
Content Management Center of Excellence in Boeblingen, Germany.

Tony Pearson
Senior Storage Consultant, IBM System Storage™, Tucson

Mia Winn
Marketing Program Manager - IBM Enterprise Content Management

Become a published author

Join us for a two- to six-week residency program! Help write a book dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You will have the opportunity to team with IBM technical professionals, Business Partners, and Clients.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you will develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks® in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  `ibm.com/redbooks`

- Send your comments in an e-mail to:

  `redbooks@us.ibm.com`

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Introduction

In this part of the book, we introduce basic storage management and enterprise content management concepts.

1

ECM summary for storage specialists

In this chapter, we present an introduction to IBM ECM strategy and solutions. Emphasis is given to the usage patterns of the data, and their storage requirements.

1.1 What Enterprise Content Management is

An Enterprise Content Management (ECM) system is designed to contain unstructured information such as files, images, and drawings. Its purpose is the delivery of the right content to the right person at the right time, and in the right context. In enterprise content management, the term *content* is used to refer to unstructured information; while structured information such as database content is referred to as *data.* Although *data* is also present in ECM systems, it is used in a supportive role to help locate and manage the *content*.

ECM systems architectures vary from vendor to vendor, but most systems include these components:

- An index repository, typically a database, which provides the users of the system with summary and search capabilities of the content stored in the system. The index data is commonly referred to as *metadata*.
- A document repository (such as a file system) where the actual documents are stored.

Also present are components that provide access to the system. End users might require direct access via Web or product-specific clients. Alternatively, users might have access to the content in the system indirectly via other applications. In this case, the applications typically connect via application programming interfaces (API) provided by the ECM system.

In some special cases, the data and content might have to be stored on the same, protected (non-rewritable) storage, to meet compliance requirements. In that case, an additional copy of the data remains available in a database to facilitate the search for content.

1.1.1 Generic diagram and functionality

Figure 1-1 displays a generic diagram of an ECM system.

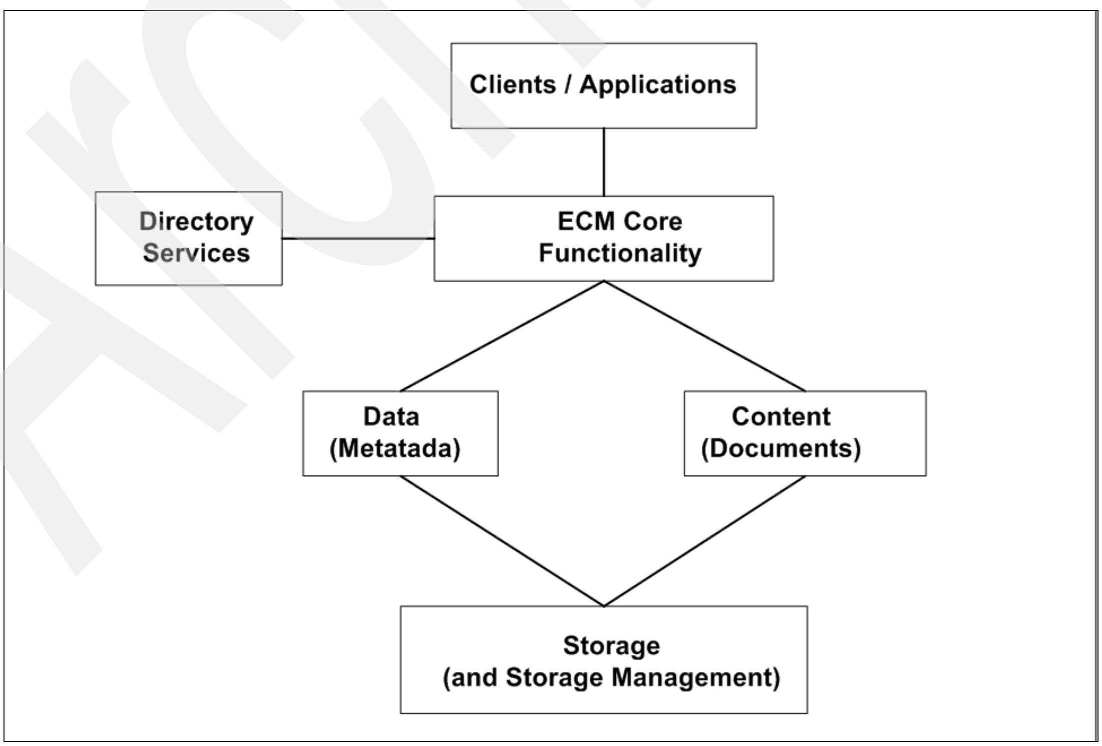


Figure 1-1 Generic ECM diagram

All relevant information is stored either as data, content, or a combination of data and content. The data component is typically a database, and its contents are the metadata or indexes on the content. The core functionality is provided by one of more applications, which are accessible to clients or other applications. Directory servers can typically be leveraged for authorization and authentication services. The content can be stored directly on the file system, or via a storage management layer.

When an end user requests access to the system via a client, the system checks the user credentials against the directory service. Upon authentication, the user is allowed access to the "data" portion of the system, and is able to browse and search for specific content within the user's authorization profile. After conducting a search, the user is presented with a hit list of content. Each item in the hit list is a metadata element that has a logical connection with content, for example a document, residing in the "content" portion of the system.

The user can then select which content they want to access by clicking on the metadata associated with it. This selection causes the (ECM Core) system to fetch the appropriate content from the "content" component, and retrieve it to the interface that the user is utilizing; for example. a Web client or via another application.

At this point, the system might flag the content as "checked out" (for example, a text document) or as "open," to avoid other users being able to save changes until the original user closes it. The original user then can view and if necessary make changes to the document, and then save those changes when finished. At that point, the system will check the document back into the system, removing the "checked out" flag.

1.1.2 Information flow

There is a particularly important aspect of this interaction, which arises from the difference in information flow:

- Volume of metadata involved: The queries for metadata are typically done against a database management system, and the metadata is designed to provide the user with enough parameters so that the hit list is short enough for a quick visual scan, for example 20 lines, or less than 1 KB in size.
- Volume of content involved: The whole document is typically delivered to the user. Document size varies dramatically, but is typically larger than 100 KB. In the case of videos, files are typically several MB in size, and can even reach several GB.

  Metadata not only allows the users to search and locate the content, but also make business decisions on the content. For example, users might decide to retrieve content after reviewing metadata elements showing that the document belongs to customer John Smith and was filed three months ago. In other words, metadata allows companies to know what documents they have stored, and is therefore of extreme importance to be available to potential users at any time, and virtually immediately.

  Given the extreme importance of the metadata availability, and its small size (relative to content), it is not surprising that this information has to be given priority in terms of storage. As a result, the fastest and most reliable types of storage, typically very fast disks, are recommended for storing metadata.

1.2 Why ECM

Enterprises have diverse business requirements revolving around content. These include the ability to access data in various repositories, utilize content in business processes, and comply with regulatory standards. These requirements dictated the evolution of ECM systems abilities to *access*, *manage* and properly *deliver* data and content, plus enable regulatory *compliance*, as discussed in the following sections.

1.2.1 Accessing data and content

The ability to *access* data and content in different repositories is of crucial importance. For example, in mergers and acquisitions, companies must access information in repositories that were not originally part of their infrastructure. Many companies also typically rely on different vendor technologies inside their own enterprise. It is therefore highly desirable that an ECM system provide appropriate access to all existing and acquired (as in the case of mergers and acquisitions) repositories, without the necessity for data and content migration, which are typically time-consuming and expensive.

This extended access, commonly referred to as *federation*, must include the abilities to create, read, update, and delete data and content, while still respecting the security associated with the content. One important advantage of federation services is the ability to hide the complexity of the information for the applications, allowing development to treat all information as if it came from a single source. Federation also greatly simplifies reuse of existing information.

1.2.2 Managing data and content

The ECM system must also provide the ability to *manage* its information, from the time it is created or ingested, to the time it is deleted. Proper interfaces for content ingestion must be provided, as well as the methods to create the indexes (metadata) associated with them.

Proper security must be provided, typically via user authentication for system access, and authorization to access the appropriate content. For example, a company might allow only a particular user group to view (read) a certain type of document. Among the users in the group, distinct levels of security could be defined: All can read the documents, some can create, others can update, and only a few can delete.

User security information is typically available in directory servers. Therefore, it is desirable that ECM systems can integrate with directory servers in order to leverage the information existing there.

Depending on the type of information involved, the ability to update data and content might be required. In addition, version control of documents is a common requisite.

1.2.3 Delivering information

Today's competitive environment is forcing companies to drive new business value from existing assets. As a result, it is highly desirable that ECM systems *deliver* information to the end users in a flexible manner. This can be accomplished with the use of Web services for information delivery. This approach fits into a services oriented architecture (SOA), allowing components to be built for a certain immediate requirement and reused across the enterprise, or an extended infrastructure that includes business partners.

1.2.4 Regulatory compliance

The increase pressure of government-mandated regulations has forced companies, particularly those publicly owned, to re-evaluate and improve their archiving and retention processes. In addition to *compliance* with the regulations, companies must consider the discovery costs associated with litigation activities.

Archiving and retention activities offer opportunities for operational cost reduction, for example by migrating inactive data to less expensive storage media. The ECM system must therefore provide capabilities to implement the appropriate policies related to the life cycle of the information.

1.2.5 ECM driving forces

At this point, it is beneficial to look at specific facts and requirements around data and content, that are driving the evolution of enterprise content management.

- **Content is stored in many different repositories**. An ECM system must be able to access and leverage all content by supporting integration and federation to multiple vendor repositories. With federation, content can be accessed and consistently managed in place, without forcing migration. Business processes and applications remain intact, without forcing application rewrites or changing user interactions. Integration and federation allow clients to innovate more rapidly by deploying new applications faster.
- **Increasing services sophistication**. Future content capabilities, such as content analysis, will be demanded by clients. The ECM architecture must be extensible so new features can be added to support the development of new offerings that bring sophisticated functionality, all with simplistic and intuitive ease of use. The importance of a platform based on industry standards, in conjunction with a strong business partner program, allows business partners to develop complementary solutions. This enables the creation of solutions that leverage expertise well beyond the ECM vendor's own capabilities. In addition, the existence and continuing support of development tools such as Software Development Kits (SDKs) and APIs and that can interact with the ECM system is of importance to customize the solution to the each customer's unique situation.
- **Innovation requires "active" content**. Streamlining business processes is critical to reducing the time, cost and risk associated with many business operations. Active content can automatically set processes in motion and drive task resolution to speed response and make businesses more agile. As a result, an ECM system must include, or be able to couple to services that offer business process management capabilities.
- **Increasing user sophistication.** Users expect more and more complex content, and also expect it to be readily and easily accessible. As companies find new ways of using content, the points of access to content and content services are increasing. Whether through direct ECM interface, portals or Content Enabled Vertical Applications (CEVAs), business consumers of ECM expect flexibility, and they expect control to leverage the power of ECM. Developer innovation will play a role in meeting these expectations. ECM systems will be expected to interface and leverage functions such as new Web 2.0 technology based interfaces and enhanced tools for business process management modeling, analytics and monitoring, search, compliance and collaboration.

1.3 Typical business uses of ECM

This section describes some common business requirements and the particular ECM system that provides a solution. A summary of the type of content, and its usage, is also included.

1.3.1 Image management

Companies might have to archive images for several reasons. For example, motorized equipment (such as cars) manufacturers in the U.S. are required to keep a record of the individual contracts of sale to end users. The contract bears the guarantee terms and the signature of the customer. The least costly solution in the short term is to keep paper records, but this becomes more costly as the volume of paper increases, occupying office space. In addition, paper records are costly to search, and are not readily available to customer representatives who might require access to individual contracts to fax to customers who have lost their originals (a common customer request). Paper is also a fragile media which deteriorates in quality over time. Finally, the contracts have a definite life cycle, at which time it would be desirable to destroy them since they are no longer active documents. The process of searching through paper documents and subsequent destruction also incurs significant cost to the companies.

The solution in this case is to scan the contracts into an image file, typically JPEG, which can be imported and managed in an ECM system. The scanning process is planned so that a cover sheet bearing proper customer and contract identification (customer name, ID, contract type, and date) is placed on top of each contract. That allows for efficient automated scanning. The information in the cover sheet is typically written in bar code, which is virtually error-free, especially when compared to character recognition techniques. The ingestion process is configured to created metadata from the bar code information, and is typically placed in a database for efficient search by end users.

Once ingested, the contracts can be made available via a client such as a Web interface. End users such as customer representatives can then search and find the contract while they talk to the customers on the phone. Some companies also allow access to the contract to their actual customers. In this case, a search restriction is placed so that each customer can only retrieve their own contracts.

The contracts can be set to expire at the appropriate time. The system administrator can be notified of upcoming expiration dates and can proceed to the actual deletion. The deletion process can be logged for auditing and record-keeping purposes.

1.3.2 Document management

Some documents are complex in nature, made up of different components, and typically require the collaboration of several resources or even teams in their creation process. One example is engineering specifications, which can require the collaboration of teams in different geographic locations. In some cases, companies even allow business partners to work on different components of the specifications, according to their particular expertise.

Such a collaborative effort requires a system where the document creation process can be specified and managed throughout the life cycle of the document. This includes access and version control for creators, editors, reviewers, and approvers, for example. Proper rules can be established for the documents as well, such as version control and use of approved templates.

Another example can help illustrate other common requirements of a document management system. A company might have an operating procedure manual that consists of a word processing file that details procedures, an embedded spreadsheet that defines the scheduling, and a computer aided design (CAD) document that displays the required equipment detail. In order to effectively manage these compound documents the system must be able to manage the relationships between all of these components, including their individual versions.

Some companies, however, still rely on ad-hoc methods for document management, typically using e-mail for file transmission and a shared drive as a document repository. This process is intensive and inefficient, and increases the possibility of security breaches.

1.3.3 E-mail management

All companies use e-mail to conduct business and communicate. E-mail systems have grown exponentially — the average number of e-mails per day easily exceeds 100 for each user, and is increasing. Also increasing is the size of e-mails and attachments. Companies require a way to effectively manage their growing e-mail, bring it under their compliance program, and search it when required for litigation. Challenges and “pain points” include these:

- **IT:** Increasing storage costs; server performance degradation; longer backup and restore times; more costly and time consuming server upgrades and consolidation; e-mail system downtime.
- **User:** Slow performance; forced to use a local or personal archive that is un-managed, not under the compliance program, is difficult to perform e-discovery in case of litigation. In addition, local archiving practices put the companies at risk since it is virtually impossible to ensure expired e-mails have been deleted, making them liable to be discovered during litigation.
- **Compliance and Legal:** Costly, time consuming and potentially error prone proof of compliance and e-discovery.

Many IT organizations have attempted to deal with these challenges unsuccessfully by limiting individual mailbox sizes and by deleting e-mail that reaches a certain age (for example, an individual user gets a maximum of 10 MB or 50 MB of space; all e-mail is automatically deleted after 90 days). However, these create regulatory compliance and legal discovery issues and risks, as well as annoying the users. Organizations have also found that e-mail backup tapes are completely insufficient for demonstrating regulatory compliance and responding to litigation discovery orders.

Many government and industry regulations mandate that information be retained for minimum periods of time. Companies have to retain information to demonstrate compliance, and to defend their conduct in any law suits. Companies must be able to:

- Identify and retain the right information for the right periods of time
- dispose of information at the end of its useful life
- apply legal holds to hold or suspend disposition during an audit or litigation
- respond to audit or litigation discovery orders, sometimes in as little as 72 hours
- demonstrate regulatory compliance through consistent execution of company policies

The proper solution is the adoption of an e-mail management system that enables companies to meet the above criteria. One of the key capabilities of such system is the enablement and enforcement of e-mail archiving policies. An example policy follows:

- All e-mail over six months old is to be archived.
- All users can retrieve their archived e-mail as required. The retrieved e-mail will be re-archived three days afterwards.
- Users belonging to the company legal and contracts departments will have all their e-mail archived, even before it reaches the users desktop client.
- Each e-mail must reside in the following storage media, depending on the age:
  - Up to six months: In the e-mail system (fast disk)
  - Six months to one year: Archived to slow disk
  - One to five years: Archived to tape
  - Older than five years: Deleted

1.3.4 SAP data archiving

SAP® databases tend to grow exponentially as the volume of business increases. The main consequences of this volume increase are:

- A decrease in performance for the end users
- Increase in database backup time, and therefore, an increase in the restore time in case of data loss
- Increased time for database data reorganization
- Increase in use of expensive storage, since the database is mounted on the SAP production system

To overcome these issues, SAP provides built-in archiving capabilities. The SAP archiving agents works by creating archive files that contain logical collections that pertain to the appropriate SAP objects. The SAP archiving facility is then able to place those archive files in the database file system. That information is shown as archived files and can be recovered by the users with appropriate authority.

This approach helps slow down the database growth, and therefore helps avoid slow performance, as well as database back-up, restore, and reorganization time. However, some business issues are not addressed:

- Increased use of expensive storage continues.
- There is lost functionality: The default SAP archiving tools do not allow users to drill down on the data, after they restore archived files. This capability is desirable to produce tax-related reports, so in case of an audit, the company might have to spend substantially in order to pull the appropriate data from the archived files.

The appropriate ECM solution must therefore address these issues. The IBM solution provides an SAP-approved interface that:

1. Scans the SAP-archived files, and moves then to a more appropriate storage repository (such as slow disk), extraneous to the SAP system.
2. Creates a link that the SAP system can use to recall the archived files in the extraneous storage location.
3. Upon successful migration of the archive files, deletes the original SAP-archived file.
4. Allows use of third-party applications specialized in SAP data archiving, instead of the SAP-supplied archiving capabilities.

Another advantage of the IBM SAP data archiving solution is the ability to import documents, such as images or word processing documents or spreadsheets, and associate them to specific SAP transactions. This enables SAP end users to have access to those documents in the context of the transaction they are accessing, from the SAP user interface.

Additionally, the IBM SAP data archiving interface can be configured to provide access to those documents from other applications. This decreases the requirement for use of the SAP system, therefore allowing saving on IT resources.

1.3.5 Report management (electronic statements)

Many companies utilize vast amounts of information that are computer-generated, commonly referred to as data streams or transaction-related content, such as customer monthly bank or telephone statements. This type of information is usually printed and mailed to customers.

The data streams and their related documents are at the heart of customers' organization's operations. They might directly touch almost every aspect of a business, including back-office analytics and reporting, e-commerce and customer service. In many cases, content distribution through print documents remains the most common — albeit expensive and inefficient — option for business users and customers.

In addition, the lack of reliable management, transformation and delivery tools for critical business content has left it distributed across an organization, causing several problems:

- Reduced user productivity related to complex search requirements, and inaccessible or unindexed data.
- Reduced customer satisfaction because of a lack of self-service capabilities and slower customer-service response times.
- Higher content-management, printing and storage costs.

To address these challenges effectively, enterprises must make their business processes as efficient as possible — and maximize user productivity — while at the same time minimizing risk of any data loss and ensuring that the company meets regulatory-compliance mandates. They also require the capability to access transactional content in easily-digested and shared forms that match user requirements and skill sets, without requiring desktop installations or training for each user.

A major aspect of business efficiency means making content readily available across and beyond the organization. An enterprise report management solution is required that avoids the inefficiencies of traditional paper-based content. Content is transformed into useful, insightful and in-context information and can serve a crucial role in making customer information available within e-billing and e-commerce customer service solutions. It can also help your customers provide access to critical back-office applications — the reports and information that companies require to manage business on a daily basis.

A typical report management solution can enable an enterprise to store all reports that are printed and mailed to customers, so that they can be available for search and reuse in an electronic repository. In addition, some companies might elect to avoid printing the statements, and instead make the reports available to their customers via a Web interface. Users can then view their reports when they wish, and print copies if required.

2

Storage for ECM specialists

In this chapter, we introduce some basic storage concepts for ECM specialists. We then outline and compare the storage building blocks that are available to build an ECM storage infrastructure.

We cover the following topics:

- Introduction to storage terminology and concepts
- Overview of how storage I/O works
- Storage products and components
- Specialized storage solutions

2.1 Introduction to storage

In computing terms, data storage — or storage for short — refers to components or devices that store data for some period of time and return it back to the application upon request. Storage can be either volatile or non-volatile, depending whether information is retained is the storage device when it is powered off.

An example of volatile storage is computer random access memory (RAM), which loses all information when powered off. Non-volatile storage, also termed mass storage, refers to devices that do not lose information when powered off. Examples of mass storage devices are disk, tape, optical devices such as CDs and flash memory devices.

We commonly use the term *storage* to denote non-volatile storage devices such as disk and tape and terms such as *storage management* and *storage virtualization* to denote software components used to manage and to optimize access to storage devices.

Computer applications use the CPU to processes data stored in RAM memory and if the power is turned off data is lost. To preserve data, applications read and write permanent data to storage devices such as disk and tape. These are called input and output (I/O) operations.

2.1.1 Storage terminology

Before we start to discuss storage architectures and storage technologies, we introduce some basic terms and concepts. The term storage denotes any media that can contain computer data when it is powered off and return it on request to the application.

Data is often stored on *disk* storage devices; these allow the application to store and access the data upon request. Disk storage devices are often grouped into *disk subsystems* that allow a certain number of disks to be managed as one single entity. There are various kinds of disk devices that offer varying performance characteristics. Often the industry refers to *SCSI* or *FC* disk and *SATA* disk storage devices, the former have higher performance characteristics and higher cost than the latter.

Storage devices can be addressed using diverse protocols; the most commonly used is the *SCSI command protocol*. The SCSI protocol defines the commands that can be sent to a device and the answers that the device can return. Storage devices are commonly addressed by logical unit number or *LUN* and relative byte address *RBA* offset. This type of IO is also termed *block-IO*.

Storage devices are either *locally attached* to a server or can be connected to one or more servers through a network. Storage area networks (*SANs*) commonly use the Fibre Channel (*FC*) transport protocol and most storage devices and subsystems offer *FC connectivity*. Storage networks can use other transport protocols such as Infiniband or even TCP/IP.

Collections of storage devices are called *storage subsystems* and offer the advantage of aggregating and managing multiple devices as a whole. Disk storage subsystems often offer functions such as *RAID* for protection against the loss of an individual disk, *snapshots* to create fast copies of LUNs and *remote replication* to send data offsite to a second subsystem for availability. Remote replication can be *synchronous* or *asynchronous*, depending on whether data is immediately sent to the remote site or if it is sent at a later time.

Disk storage is more expensive compared to *tape storage*. Tape storage consists of *tape drives* that are used to read and write *tape cartridges* that actually contain the *tape medium.* Tape drives and tape cartridges are often contained in a *tape library* that provides storage space for the cartridges and also contains a *robotic mechanism* for moving tape cartridges from storage slots to tape drives and back under the control of an application.

2.1.2 How storage I/O works

In order to help you understand the options and enablers that are available in the storage stack, we now discuss how data is stored onto storage devices. Figure 2-1 shows the various layers that are involved in completing a storage operation.

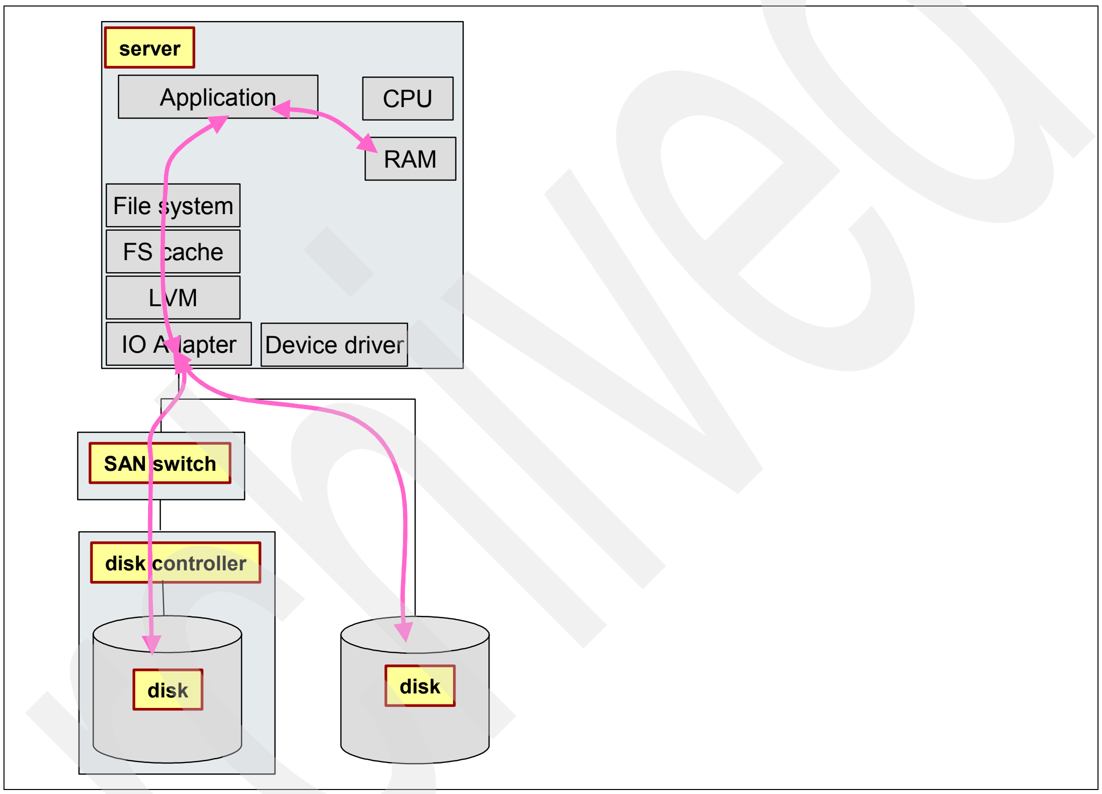


Figure 2-1 IO path overview

An I/O request is initiated at the application layer and traverses multiple abstraction layers:

- File system and the file system cache
- Logical volume manager (LVM)
- I/O adapter or HBA
- Optionally a storage area network (SAN) switch and disk controller
- A storage device such as disk

2.1.3 File system

The file system is an abstraction layer because it hides underlying storage complexities. It allows you to name data items based on content name, the file name, and later retrieve the content based on its name without having to remember the physical location of the data item on the storage device. A file system simplifies application access using named objects in naming scheme that uses concepts such as directories, file names and file name extensions.

As an example, we look at the fully qualified file name */mydir/subdir/file3.4me*, in which the directory name is */mydir/subdir/*, the file name is *file3* and the extension is *4me*.

File systems translate file names and offsets to block addresses in LVM. A file system inventory of files, or file metadata, is maintained in an index structure. Common metadata structures are the inode table in UNIX file systems and the master file table (MFT) in the NTFS. The file system metadata contains the file name to LVM block address that stores the actual file data. A file can be mapped to one or more blocks. File system metadata can also contain useful information such as file access permissions and modification and last reference dates.

2.1.4 File system cache

Common file systems also offer a function called file system cache that stores recently read or accessed data. File system cache provides fast reuse because it avoids repeated I/Os for data that is already in the cache itself. The logical volume manager (LVM) layer, shown in Figure 2-2, performs various volume management functions inside the server.

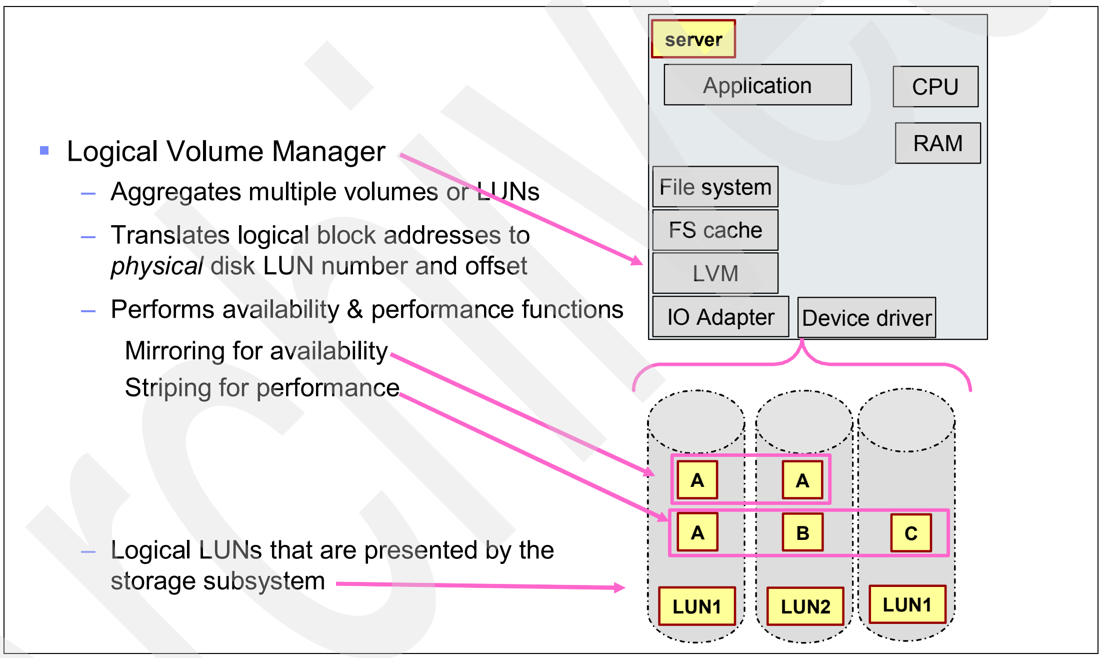


Figure 2-2 The logical volume manager (LVM) functionality

2.1.5 Logical Volume Manager

LVM sits below the file system layer and above physical IO adapters. The LVM layer translates a logical block request from the file system into physical disk requests using logical unit number (LUN) and offset inside the LUN itself. The underlying LUNs can correspond either to physical disk devices or to logical representations of LUNs inside a storage subsystem or a block storage virtualization appliance.

Most enterprise class LVMs offer storage availability and performance functions. LVM mirroring can replicate the same physical data between different LUNs for availability and stripe data across physical LUNs for performance. Some LVMs might also implement RAID functionality; in this case, we speak of a software RAID implementation.

2.1.6 I/O adapter

When the I/O operation has traversed the LVM layer, it is passed to a device driver layer and exits the system from an I/O adapter, as illustrated in Figure 2-3.

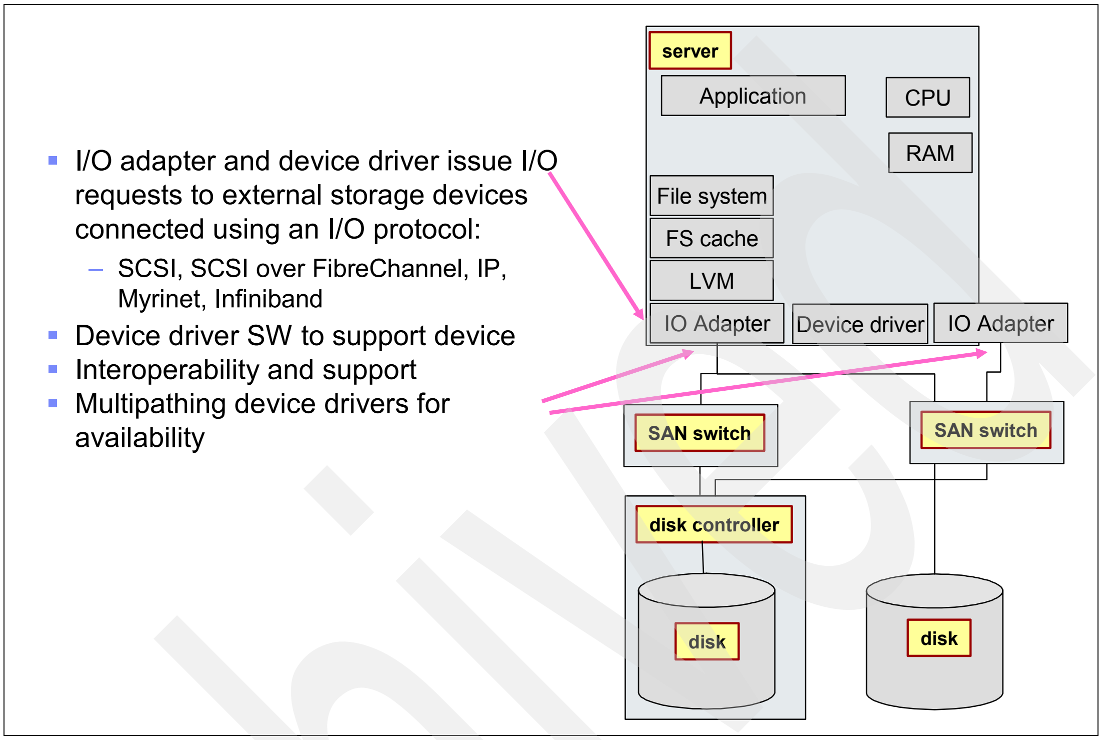


Figure 2-3 The device driver and IO adapter and storage network level

Device drivers are used as an interface between the LVM layer and an underlying storage device. The device driver is usually supplied by the hardware vendor and acts as a translator between the LVM requests and low-level device dependent instruction codes required by a specific device in the form of SCSI commands. The device driver sends the SCSI commands to physical I/O adapters where it is encoded into electrical or optical signals and sent over a transport link using protocols such as Fibre Channel (FC), IP, Myrinet, and Infiniband.

Some device drivers offer multipathing functionality for availability to transparently failover and continue I/O requests on remaining paths if one of the paths to the storage device fails.

Once the I/O request has exited the I/O adapter and left the server it traverses the network to reach the storage device. The layer between the server and the storage device is called the storage network fabric.

2.1.7 Storage network fabric

The storage network fabric consists of interconnected storage networking devices such as switches and gateways. SAN switches offer any to any connectivity, which means a request on any input port can exit on any output port, unless restricted by switch security mechanisms. Storage networks are commonly based on the Fibre Channel (FC) storage protocol, but other protocols such as Myrinet or Infiniband can be used in specific implementations. The TCP/IP transport protocol can also be used to transport storage requests, iSCSI is an example of such an implementation where SCSI commands are sent over an IP network.

To avoid single points of failure, component redundancy is usually implemented. From a storage perspective, servers are usually configured with dual redundant adapters connected to two independent, non-interconnected, SAN fabrics. Storage subsystems are connected to the redundant SAN fabrics using multiple ports.

SAN security and LUN masking

Because multiple storage devices and servers can be connected to the SAN, access security is required to ensure that servers only access LUNs intended for them. SAN security is implemented with storage LUN masking and SAN switch zoning and virtualization. LUN masking is often implemented at the storage subsystem layer and ensures that a particular LUN can only be accessed by a specific set of server HBAs. I/O requests originating from different HBAs are refused. Switch zoning allows requests coming from a specific switch port or SAN connected device port to access only a subset of outgoing switch ports. High-end SAN switch vendors offer SAN virtualization function where an enterprise switch or director can be logically partitioned into smaller switches.

The I/O request then exits the SAN fabric and is sent to a storage device, or in the case of locally connected storage devices, the I/O request is sent directly to the device. Figure 2-1 on page 15 shows the I/O being sent to a disk storage device, but the general functional flow does not change much if the device is a tape or an optical disk.

2.2 Storage products and components

There are multiple kinds of storage devices. We often hear terms such as disk and tape, optical storage, and flash drives. Why do so many technologies exist on the market and what are the relative advantages and disadvantages of each technology? We do not discuss all available storage technologies, but rather concentrate on the most commonly used and widespread ones.

Different storage technologies have different uses and applications. One common and easily understandable example is tape backups of disk storage data. Data on disk can be accessed fast because the disk is spinning and only the disk head has to be positioned to where the data resides, and then the data can be read. This time is called the locate or *positioning* time.

Data is backed up to tape so that it can be recovered if some accident happens to the disk data. Tape is used as a backup medium, and not disk, because tape is deemed to be more cost-effective than disk. The lower cost allows you to keep multiple redundant copies of your disk data. Tape devices are also termed removable devices because the tape cartridges that actually contain the data have to be physically moved from a storage location to a tape drive and the tape has to be unwound from the reel and positioned to where the data is stored. These are respectively called the *mount* and *positioning* times.

In small systems, storage devices are locally attached, whereas larger installations prefer to share usage of multiple devices so they connect the devices to some kind of networking layer such as a SAN.

Other storage media such as optical devices are also available on the market and are used in certain cases where you require media that is not physically modifiable. This type of media is used less and less.

2.2.1 Storage performance indicators

Storage devices are characterized by a set of parameters that give an indication of how well each device will perform in a given environment or under certain conditions. Cost is an obvious parameter and cost comes as a trade-off of performance. As with cars, faster cars are often more expensive than the more sedate ones. Some important performance characteristics of storage devices describe how fast they can store or retrieve data for a given application request. The main performance indicators are:

- **Mount time:** The time required to retrieve a tape or optical media and insert it into a reader such as a tape drive. This parameter is not applicable for disk storage. The smaller the time is, the better.
- **Positioning time:** The time required to locate the start of a particular data item on the physical media. The smaller the time is, the better.
- **Sequential data rate:** The amount of data per unit of time that can be transferred to or from the device. The higher this amount is, the better.
- **Random I/O Operations (IOPs) per unit of time:** The number of operations that store or retrieve a fixed amount of data that can be handled by a storage device in a given amount of time. The higher this amount is, the better.

Table 2-1 compares the overall characteristics of different kinds of storage devices.

Table 2-1 Comparing storage component characteristics

	SCSI disk	SATA disk	Tape	Optical
Mount time	N/A	N/A	High	High
Positioning time	Low	Low	High	Medium
Sequential data rate	High	High	Very high	Medium
Random IOPs	High	Medium-low	Very low	Low
Estimated TCO	High	Medium	Low	Medium

Tape has traditionally been the repository for backups and archived data, it can also be used by ECM applications either for infrequently used data or for large sets of data that have to be read sequentially. Tape is interesting because its total cost of ownership (TCO) is low. Tape does not spin when not in use, so it uses no power. Optical technologies, common a few years ago, are used less frequently today.

Enterprise-class SCSI disk is often used by the most critical workloads, whereas SATA disk are a common repository for ECM data when large data volumes and long retention periods are required.

2.2.2 Disk storage

The term *disk storage device* can refer to one of two things: a standalone disk drive; or a disk storage subsystem that contains many individual disk storage devices, as illustrated in Figure 2-4.

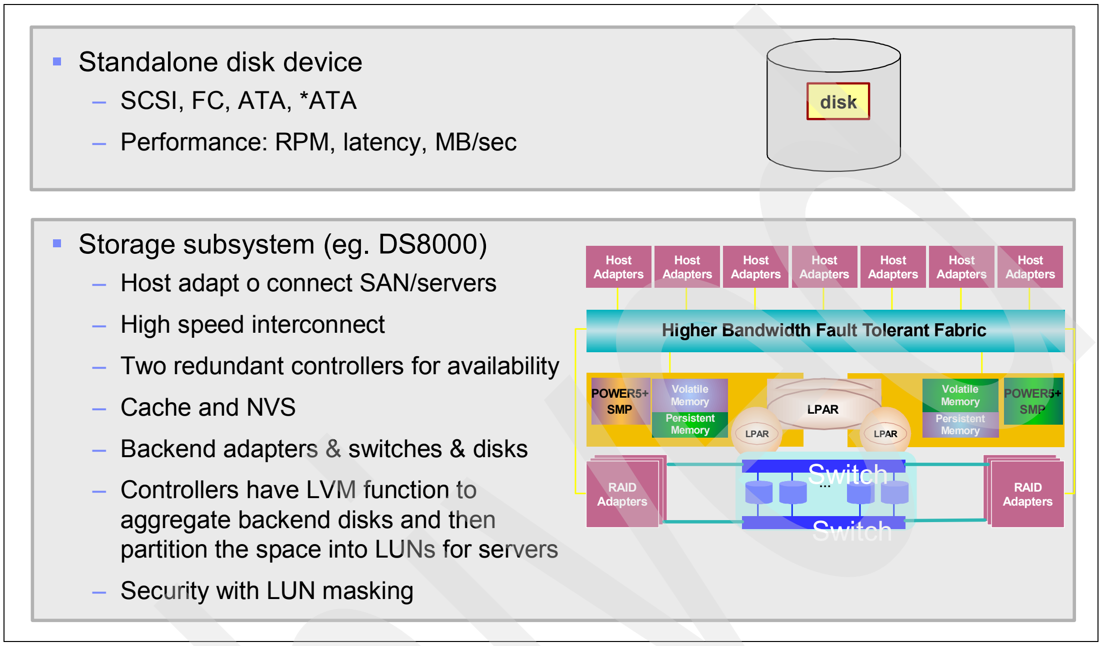


Figure 2-4 Disk devices and subsystems

Disk drives are composed of a rotating disk coated with a magnetic medium and a head that moves across the disk to read and write data on or off the magnetic surface. Disk storage subsystems contain many individual drives that are aggregated together with RAID techniques for protection against individual drive failure and to improve performance. They usually contain dual controllers for redundancy, cache memory to improve the response times for repeated requests to the same data, and access security features to allow partitioning and allocation of storage to multiple servers.

Disk drives are characterized by capacity and performance indicators that depend on factors such as drive rotational speed and head movement and positioning times. Capacity is easily understood and reflects the amount of data that can be stored on a specific drive. The drive rotational speed tends to improve sequential data rate, whereas head movement impacts the number of requests that can be made to a disk device in a given amount of time.

There are various types of disk on the market, and often a distinction is made between FC and SATA disk, as illustrated in Figure 2-5.

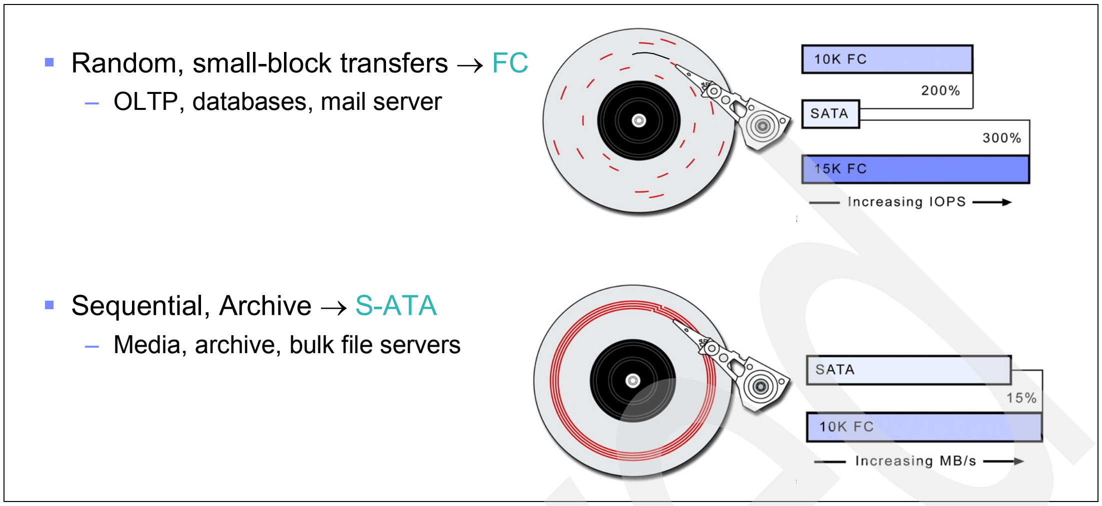


Figure 2-5 Fibre Channel or SATA disk

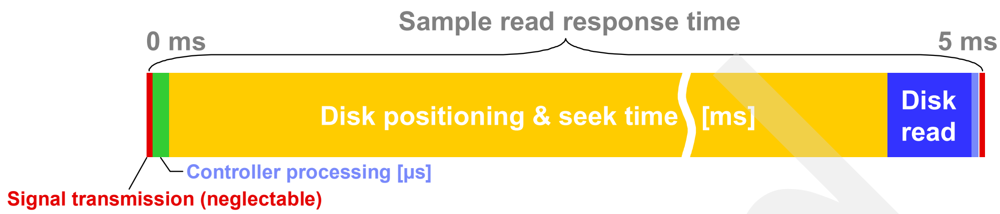
Fibre Channel (FC) disk has better performance characteristics and has to be chosen for the most demanding workloads. Disk prices relate to the mechanical quality of components. 15.000 RPM drives require expensive bearings. ATA drives use cheaper bearings, so SATA drives are built for lower duty cycles than FC drives. SATA drives are built for 20% duty cycle, whereas SCSI / FC drives for 100% duty cycle. While the disk connector or plug types differ, the major difference between these two types of disk drives is mechanical longevity, and cost.

FC drives are recommended for storing database type data such as the ECM metadata database, whereas SATA drives can be appropriate for storing ECM content. Multiple disk technology tiers can satisfy different application requirements. We often use the term storage tiers to denote multiple sets of storage devices grouped by homogenous cost and performance characteristics.

Disk drives and disk storage subsystems are addressed at a block-I/O level as LUNs. Disk storage subsystems are usually connected to a FC SAN so that the disk space can be partitioned between multiple servers. High end storage subsystems gain their speed from multiple aspects such as:

- Large read cache with prefetch mechanisms that improve data rate
- Write cache that is battery backed and mirrored improves write I/O rate
- Multiple striped disks that offer improvements in data rate and I/O rate
- RAID processing in dedicated hardware that offloads the server
- Dedicate processor resources for advanced functions such as mirroring
- External storage that can be shared between multiple servers, allowing fast application failover

Contrast this with internal disks, which can only be optimized through appropriate placement of data, usually by putting it on middle tracks. External storage is often faster than internal disks, as illustrated in Figure 2-6.



	Internal disk	External controller
Processing	*On internal adapter*	*Powerful extra processor*
Seek time	*Data layout related*	*Reduced due to cache*
Disk read	*Single disk*	*Multiple striped disks*

Figure 2-6 Comparing external and internal storage characteristics

The performance increase is due to the functions offered by the external storage subsystem, such as cache and processor offload.

Disk storage subsystems also offer manageability and scalability benefits. Multiple storage devices can be consolidated into a smaller set of shared subsystems and capacity can be reallocated between applications as required. Subsystems are also easier to manage and protect than small individual storage devices and offer advanced management functions and redundant components.

RAID protection schemes are implemented in storage subsystems and offer increased levels of availability and performance. There are various types of RAID, denoted by a number:

- RAID 5 is commonly used and is generally a good choice. It offers a space efficiency factor of 87.5% with 12.5% used for redundancy in an 8-disk RAID group.
- RAID 10, mirrored RAID 5, is used for heavy random write workloads and has a space efficiency of 50%.

  RAID 10 is advisable if you want to avoid long “hot spare” rebuild times with RAID 5.
- RAID 3, fixed parity, has benefits in some streaming media applications.
- RAID 1 is one-to-one and lacks the advantages of striping over many disks, so it is not recommended.
- RAID 0 is striping without redundancy and does not offer security.

IBM System Storage has a comprehensive range of disk storage products, the portfolio covers the requirements of a wide range of implementations, from entry-level systems to large enterprise solutions. This is called the IBM System Storage DS™ family. It consists of:

- DS3000 series for small to medium deployments
- DS4000™ series of mid-range systems
- DS6000™ and DS8000™ series enterprise servers

IBM also offers a full range of IBM System Storage capabilities such as advanced copy services, management tools, and virtualization services, which are available to help protect data.

2.2.3 Tape storage

Tape storage devices are usually more cost effective than disk, comparing cost per unit of capacity, when medium to large data volumes have to be stored. Current tape storage devices offer native capacities close to one TB per tape cartridge, so a library with 1000 cartridges can contain around 1PB of data.

Tape storage devices are used to store inactive data, data that is infrequently accessed by applications or large data sets. This is due to the nature of tape devices that have to be mounted into a tape drive and positioned to the start of the data, operations that typically take approximately one to two minutes. After this amount of time, the tape drive starts streaming data back to the application; in the case of enterprise class tape drives, the data can be streamed at high data rates of about 100 MB/sec per tape drive. For small amounts of data, tape mount and positioning time can be a considerable percentage of the data access time; whereas for large amounts of data on the order of tens or hundreds of GB, the mount time becomes negligible compared to the total data transfer time.

Some high end tape devices offer functions such as data encryption to prevent the data from being read if the tape falls into the wrong hands. Tape devices also offer WORM functionality that is designed to provide a non-alterable, non-rewriteable media.

From an ECM perspective, tape is well positioned to contain data that will probably never require to be accessed, such as content archiving for legal discovery reasons or historical records. If there is a high probability of accessing certain data types, then consider the use of disk to minimize access times.

2.2.4 Tape libraries

Tape storage devices usually consist of a tape library that contains a certain amount of slots for tape cartridge storage, drives on which to mount the cartridges and a robotic mechanism to mount and unmount cartridges from the drives. A particular cartridge can usually be mounted in any available drive and multiple cartridges can be mounted in separate drives and read or written concurrently. If you require additional throughput, you can add drives to the library. Multiple drives also offer a security mechanism because if one drive fails the tape cartridge can be mounted on an available tape drive. Tape libraries are necessary for automated and unattended operations because they do not require human operators and do not suffer from the delays that could be had with a human operator.

High end tape libraries offer greater levels of scalability, reliability, availability, and serviceability. They commonly offer barcode scanners for reading tape labels and support for tape input output stations to easily insert and remove multiple tapes from the library. This is useful to extract tapes for remote vaulting. This type of library also offers greater flexibility because they have a modular design and can be easily scaled up by adding components and enclosures.

High end tape libraries also offer sharing functions that allow a library to be shared between multiple independent hosts where each system thinks it actually owns a dedicated tape library.

IBM offers the IBM System Storage Tape Systems (TS) family of products that include a comprehensive set of tape drives, tape libraries, and tape virtualization products.

2.2.5 Storage networking

Storage networking technologies allow multiple sets of servers the ability to access and share multiple storage devices through a network layer that enables more flexibility in storage access and enables any-to-any connectivity. There are many types of storage networking infrastructures because there is no single approach to solve all storage networking requirements and problems. There are various trade-offs to be made when selecting a storage networking technology such as cost, performance, maturity, and ease of use.

There are two common approaches to implementing a storage network — it can either be implemented by using Fibre Channel (FC) storage area network (SAN) connectivity, or by using TCP/IP protocols over ethernet LANs. The term *SAN* is commonly used to denote a storage area network based on the FC protocol and not to denote networks based on the TCP/IP protocol, even though the latter might be networks dedicated to storage access. From a server perspective, a storage network can be viewed as an extension to the storage bus. Data is accessed on a storage network using block level I/O requests that are sent directly from the server to the storage device. The major benefits offered by SAN are:

- Better application availability, because storage is independent of servers and accessed through redundant paths
- Higher application performance, because storage related tasks are offloaded to the storage server
- Backup processing that can be off-loaded from servers and moved onto separate devices, as with LAN free backups
- Centralized and consolidated storage for easier management and added flexibility
- Data transfer and vaulting to remote sites, because a SAN can offer enhanced storage connectivity to remote locations
- Facilitates centralized management, because it enables consolidation of multiple devices to a smaller set of large devices

Recently storage networks based on the Ethernet have been emerging and these are used either to functionally replace an FC SAN or extend an FC SAN over geographical distances. Protocols used to extend a SAN over geographical distances, besides FCP, are Fibre Channel over IP (FCIP) and Internet Fibre Channel Protocol (iFCP). Servers still use FC HBAs to access a SAN and the SAN is extended over an IP connection to a remote SAN using the FCIP or iFCP protocols.

The Internet SCSI (iSCSI) protocol instead allows a server to connect to a storage device and send SCSI commands using a LAN connection. Both the server and the storage device have to be connected to the LAN. This protocol offers great connectivity flexibility but is relatively CPU intensive on the server side because all SCSI commands must be encapsulated into TCP packets.

FC SAN storage networks are implemented with networking elements such as FC routers and switches, directors, and gateways. These are functionally similar to IP networking devices but offer Fibre Channel connectivity instead of Ethernet connectivity. FC SAN storage networks also offer virtualization capability.

IBM offers a complete set of SAN switching devices and infrastructure components. IBM System Storage SAN switches offers a complete set of devices and infrastructure components to build SANs.

2.2.6 Storage virtualization

SAN storage virtualization is used to address the increasing complexity of deploying and managing storage. Addressing the complexity also helps to reduce costs and better exploit the benefits offered by a SAN.

SAN Storage virtualization

SAN Storage virtualization enables data sharing between multiple users. It helps to deliver higher availability and better performance and can also provide varying degrees of disaster tolerance and offer increases in performance. Virtualization also allows for the aggregation and consolidation of multiple resources so that the servers above the virtualization layer see it as a whole. Virtualization can also provide policy based migration and other benefits that do not automatically result from the implementation and use of individual SAN hardware components.

SAN storage virtualization works at the block I/O level. It is usually implemented with a virtualization appliance such as the IBM San Volume Controller that is positioned in the data path between the server and the underlying storage arrays and subsystems. This kind of implementation is termed an in-band virtualization appliance. The storage LUNs in the underlying storage arrays are assigned to the virtualization appliance. They are called the managed LUNs. The managed LUNs (MLUNs) are aggregated into pools of capacity and from this aggregated capacity a new set of LUNs called virtual LUNs or VLUNs is derived. VLUNs are then assigned to the servers. The space for VLUNs is often striped across multiple MLUNs that can reside in the same or different storage subsystems, and striping is good for performance and for eliminating hot spots.

SAN storage virtualization devices also offer functions such as remote data mirroring between two different virtualization appliances or instant snapshot copies inside the same virtualization appliance as a way to perform rapid backup operations. An additional benefit of using an in-band virtualization appliance is that new storage can be provisioned and data migrated to the new storage devices without impacting data and application availability. An interesting side effect of this feature is that data storage LUNs can be moved between storage tiers transparently to the application while the application is active. For example, a test environment deployed on mid-tier storage devices can be transparently relocated to enterprise class production storage devices.

There are various other kinds of storage virtualization. At the FC SAN level, high end switches and directors offer functions to virtualize SAN fabrics and allow one switch or director to appear as if it were multiple switches or directors.

File system layer virtualization

At the file system layer there are multiple virtualization implementations. Grid and parallel file systems offer seamless access to data. This can offer advantages such as having a global namespace to access data stored on multiple storage servers and across multiple locations to parallel file systems that allow data to be accessed and stored through the use of multiple file data servers.

2.2.7 Network attached storage

Traditional network attached storage (NAS) devices are attached to a TCP/IP-based network (LAN or WAN), and accessed using CIFS, NFS or specialized I/O protocols for file access and file sharing. The data traffic is based on a file level I/O. The NAS device receives an NFS or CIFS request from a server and has an translates that request to the block-I/O commands to access the appropriate device only visible to the NAS product itself.

So a traditional NAS offers a file system that can be accessed over the LAN network and shared between multiple servers. NAS devices offer the following advantages:

- Simple installation and setup, because they do not require a dedicated SAN network and can be rapidly connected to an existing IP network
- Simple access from the host, because generally you do not require additional device drivers or software to access a NAS device
- Pooling of resources inside the NAS appliance, meaning that resources can be easily assigned between the servers accessing the NAS device
- NAS appliances also providing file sharing between multiple servers that access the NAS using CIFS and NFS

High end NAS devices offer a set of additional functionality that eases system management operations and allows enhanced levels of data protection. As an example of functionality, we discuss what function is offered by IBM N Series NAS devices:

- The SnapShot function allows for creating rapid and space efficient copies of data without impacting application availability. Individual files can be read from the snapshot, and the SnapRestore® function allows for rapid recovery of data to the original volume.
- The SnapMirror® function allows you to replicate data between to different remote NAS systems for disaster recovery purposes.
- The advanced single instance store (A-SIS) functionality offers data deduplication allowing for identical data to be stored only once, thus conserving space.
- The SnapLock® feature provides disk-based protected storage. Files that are written to a SnapLock volume cannot be modified or deleted until they reach their expiration date.

SAN and NAS storage devices are converging. Today high end NAS storage devices such as the IBM N Series offer the benefits of SAN-attached storage because they can be connected to the SAN and present LUNs. The LUNs that are presented can be accessed using the FC protocol over a SAN connection or through the iSCSI protocol lover a LAN connection. LUNs inside an N Series NAS system can also be protected using the local SnapShot and remote SnapMirror functions.

2.2.8 Storage management software

Storage management software enables you to manage data objects from the moment they are created to when they are no longer required. This functionality applies most often to files residing on servers but can also be applied to other data objects such as databases or even whole volumes. Storage management operations that can be performed on data objects might include one or more of the following aspects:

- Backup and recovery of data to protect against accidental loss or corruption, both physical loss where the media breaks or is destroyed, and logical loss where the data is intentionally or accidentally corrupted
- Archiving data objects so they are moved from primary storage to secondary storage devices and space is cleaned up on primary storage devices
- Creating offsite copies of data for disaster recovery
- Retaining data for long periods of time in non-erasable non-rewriteable storage

IBM Tivoli Storage Manager

IBM Tivoli Storage Manager (TSM) provides a comprehensive solution focused on the key data protection and management activities of backup, archive, recovery, space management, and disaster recovery. Tivoli Storage Manager allows you to separate the backup, archiving, and retention of data from storage-related aspects of the data, in addition to many other services. Tivoli Storage Manager offers various storage management functions that can be used by ECM systems to protect and manage the data throughout it's lifecycle:

- **Data archiving:** This defines how to insert data into the data retention system. Tivoli Storage Manager offers a command line interface to archive and back up files and a C language application programming interface (API) for use by content management applications.
- **Data retention:** In TSM, this defines how long to keep the data object, not the individual tape. TSM offers various data retention options, such as the amount of time to keep for archived objects and the number of versions and retention for backup operations. Data can be stored in different storage repositories based on rules such as the retention assigned to the object.
- **Storage:** This defines on which storage device to put the object. Tivoli Storage Manager supports hundreds of disk and tape storage devices and integrated hierarchical storage management of stored data. You can choose the most effective storage device for your requirements and subsequently let the data automatically migrate to different storage tiers and specify that data in TSM must be protected with additional copies to keep onsite or send offsite.
- **WORM functionality:** Also known as non-erasable non-rewriteable storage, WORM is offered by System Storage Archive Manager (SSAM), a special version of TSM used for protected storage. The Tivoli Storage Manager administrator cannot accidentally or intentionally delete objects stored in Tivoli Storage Manager. TSM offers various data retention options in SSAM, such as:
  - By date specifies the duration to retain the data.
  - Event-based determines retention on notification of a future event.
  - Deletion hold prevents deleting an object even after its defined retention period.
- **Storage management services:** These are provided by Tivoli Storage Manager. These additional storage management services facilitate hardware replacement and disaster recovery. Tivoli Storage Manager allows for easy migration to new storage devices when the old storage devices require replacing, and this will likely happen when data is retained for long periods of time. Tivoli Storage Manager also offers functions to make multiple copies of archived data.
- **Security:** This is offered by TSM at many levels. Clients and applications that connect to TSM must authenticate themselves and data contained in TSM. TSM offers transparent encryption functionality to make data secure. It can be performed either at the client, before sending data on the network or at the server because it supports tape encryption and can perform key management functions for both kinds of encryption.

Tivoli Storage Manager offers a strong and comprehensive set of functions that you can exploit to effectively manage archived data. You can consider Tivoli Storage Manager an abstraction or virtualization layer between applications requiring data retention or storage management services and the underlying storage infrastructure.

2.2.9 Storage monitoring and configuration management software

Having multiple storage devices and SAN connectivity elements makes it necessary to have a comprehensive and homogenous management solution that can help you to monitor and manage the storage environment from a single point of control. The monitoring component must be able to produce reports, alerts, historical trend analysis, and facilitate configuration operations. Conceptually, there are four levels that require monitoring and management:

- Server level
- SAN connectivity fabric level
- Storage subsystem level
- Storage subsystem replication level

Server level

You have to manage storage as it is seen from the server level or layer, including file systems, volume managers, and database management systems. You must be able to obtain current capacity and utilization reports, and to use historical trends to project future requirements.

SAN connectivity fabric level

Managing the SAN means that you must be able to discover and view the SAN network topology and receive SAN alerts. SAN management software also offers configuration management interfaces for heterogeneous switches.

Storage subsystem layer

Software components for monitoring and managing the storage subsystem layer can provide subsystem capacity, utilization, and performance data, and allow you to create LUNs and assign them to servers.

Storage subsystem replication layer

Storage subsystem replication management software allows you to easily set up replication of LUNs between multiple storage subsystems and monitor replication progress and status. Replication software is also used to ensure the integrity of data being replicated between storage subsystems.

TotalStorage Productivity Center

IBM offers the IBM TotalStorage® Productivity Center (TPC) product family to cover these requirements. TPC provides monitoring and management of storage components and devices from a single point. The TPC family consists of four components:

- IBM TotalStorage Productivity Center for Data
- IBM TotalStorage Productivity Center for Fabric
- IBM TotalStorage Productivity Center for Disk
- IBM TotalStorage Productivity Center for Replication

2.3 Specialized storage solutions

Hardware storage devices can be integrated together with software storage components to provide intelligent storage and storage management solutions. Software can be seen as an additional layer that sits on top of multiple types of underlying hardware devices and manages them as a seamless whole. Specialized storage solutions use software components to offer functions such as storage tiering and automated data movement based on customer specified policies and rules.

Having an additional intelligent management layer for storage offers the possibility of creating added functionality. Mistaken sometimes as adding complexity, it would be more appropriate to call it an enablement layer for obtaining additional flexibility. This flexibility is the enabler for better TCO, because a given storage configuration can be expanded in multiple ways, for example, by adding storage to a specific tier that requires more capacity or reutilizing existing storage devices that currently exist in your environment, either disk or tape.

This additional layer of flexibility and openness has to be contrasted with storage solutions that offer only one type of storage, the one size fits all, and do not allow you additional flexibility to adapt and change because of new and changing business requirements.

2.3.1 Tiered storage and hierarchical storage management

Data access and use patterns can change over time; the use of different storage tiers enables you to place the data in the optimal location based on usage patterns. For example, you might have high performance disk and lower performance low cost devices and initially place data on the high performance devices and later move it to the next device tier. Tape can also be used in a tiered storage environment and is often a good choice for very infrequently accessed data.

Hierarchical storage management

The function or mechanism to control data positioning and movement between the storage tiers is called hierarchical storage management (HSM). HSM functionality is characterized by the following features:

- A migration mechanism to move data to the next storage tier
- Transparent access to data that has been migrated either by recalling it to the original tier or transparently accessing the data on the new storage tier

The migration process can either be initiated by the user or application using commands, or managed by an automated process that monitors primary space utilization and initiates a migration process when predefined utilization thresholds are exceeded. The HSM system usually allows you to define policies to select which files are to be candidates for HSM.

HSM function is usually implemented either as an extension to a file system or as an integral part of the filesystem itself. IBM offers both kinds of functionality.

HSM as a file system extension is offered by the IBM Tivoli Storage Manager for Space Management product (TSM HSM). This allows you to send files that migrate to a Tivoli Storage Manager server for storage on any storage device connected to the TSM server.

General Parallel File System

The IBM General Parallel File System™ (GPFS™) is a high performance shared disk file system that provides fast access to a common set of files from two or more computers. GPFS provides online storage management, scalable access, and integrated information lifecycle tools capable of managing petabytes of data and billions of files. GPFS provides for information lifecycle management (ILM) by using multiple disk storage pools mapped to different storage types and policy-based file management rules that control allocation, movement, and deletion of data.

A file in a GPFS filesystem maintains its path and name regardless of where it is placed by GPFS policy based data management, so the application does not have to track file name changes. The application can transparently access the file regardless of where the GPFS filesystem has moved the file. GPFS can also be extended to support migration of data to tape devices under control of TSM by using Tivoli Storage Manager for Space Management.

This functionality is of interest in ECM environments because it allows you to create a seamless multi-tier storage solution that can contain various kinds of disk tiers and tape tiers and data can be moved from one tier to the other transparently to the ECM application.

Grid Archive Manager

Another IBM solution, based on a CIFS/NFS file system interface, is the IBM Grid Archive Manager GMAS solution based around the IBM Multilevel Grid Access Manager (GAM) software. GMAS is an automated, self optimizing distributed grid storage solution. It allows multi-campus enterprises to link disparate storage systems together and optimize utilization while offering full system redundancy and ensuring that multiple copies of data are geographically separated. GMAS allows for Hierarchical Storage Management/Information Lifecycle Management (HSM/ILM) based upon a file's metadata.

2.3.2 Protected storage solutions

The term protected storage is used to denote storage devices that do not allow data stored on them to be altered or deleted. These devices are intended to help support the long term retention of reference data and meet the requirements of regulatory bodies worldwide, when there is a business requirement for protected storage and also as a security mechanism and failsafe against human actions such as errors.

There are two kinds of protected storage solutions:

- Physical WORM devices that rely on physical properties of the media to avoid data being modified or deleted
- Logical or software WORM devices that rely on software or microcode mechanisms to stop data being deleted before it reaches a predefined retention date

Examples of physical worm devices are optical disks and CD-R and DVD-R media. These are no longer in common use because their performance characteristics are not as satisfactory as those of current magnetic media.

IBM offers multiple hardware software WORM solutions to protect critical data, solutions that can be used separately or connected together:

- The IBM N Series SnapLock feature
- The IBM System Storage Archive Manager software
- The IBM DR550 solution
- The IBM TS1030 and TS1040 LTO tape drives and the IBM TS1120 tape drive with WORM enabled tape cartridges

SnapLock function

The IBM N Series SnapLock function is a data function designed to deliver high performance and high-security disk-based file-level locking or WORM functionality on both near-line and primary IBM System Storage N Series storage. The SnapLock function can help manage the permanence, accuracy, integrity, and security of data by storing business records in an inalterable form and allowing for their rapid online accessibility for long periods of time. There are two versions of SnapLock:

- **SnapLock Compliance:** For strict regulatory environments, in this case, the N Series administrator *cannot* delete SnapLock protected data before its retention period expires, nor can the initially-set retention period be shortened.
- **SnapLock Enterprise:** For environments with less stringent regulatory restrictions, in this case, a credentialled N Series administrator could shorten a document's retention period to simply apply a corrected or a revised retention period, in order to to allow earlier deletion of the SnapLock protected content.

System Storage Archive Manager

IBM System Storage Archive Manager (SSAM) software is a version of Tivoli Storage Manager that has data retention protection enabled. This function ensures that objects that have been archived will not be deleted from the Tivoli Storage Manager server until the retention policies set for each object have been satisfied. SSAM actively inhibits deletion of unexpired objects. Apart from data retention protection, a SSAM server is functionally equivalent to a TSM server and offers support for a broad range of storage devices such as disk, tape, and optical media. Data can be sent to SSAM servers either with a TSM command client that allows the archival of files from a file system to SSAM or it can be sent directly from an application to the SSAM server through the use of a public release TSM API interface.

IBM System Storage DR550

The IBM System Storage DR550 is a secured and certified hardware solution based around IBM SSAM software, a server and disk storage.

System Storage DR550 offers scalable data retention solutions to store, retrieve, manage, share, and protect regulated and non-regulated data. DR550 offers secure archival and retention, tiered storage support, synchronous and asynchronous replication capabilities. The DR550 is based around IBM SSAM software, a server and disk storage. The DR550 offers:

- Provide pre-configured, integrated hardware and software solution to store, retrieve, manage, share, and protect regulated and non-regulated data.
- Offer advanced data protection options such as data encryption and policy enforcement.
- Offer a broad suite of software features for policy- and event-based data management.
- Provide optional encryption for data on its physical disk and attached storage devices (for example, tape).
- Offer automatic provisioning, migration, expiration, and archiving capabilities.
- Provide the option to use advanced WORM tape to store data objects.
- Provide a high-availability option designed to avoid single points of failure.
- Provide optional synchronous and asynchronous data replication between two DR550 systems

The IBM System Storage DR550 and DR550 Express solutions integrate a range of technologies as pre-configured solutions. These solutions provide upgrade options for connectivity, storage capacity and additional external tape or optical storage to petabytes of storage per system. These solutions support the ability to retain data without alteration throughout their designated retention period.

Data is stored in the DR550 in the same way as it is stored to a SSAM server, either with a TSM command line client that allows the archival of files from a file system to SSAM or it can be sent directly from an application to the SSAM server through the use of a publicly released TSM API interface. An additional feature called DR550 File System Gateway offers file archiving capability without requiring any application-based enablement, and provides Network File System (NFS) and Common Internet File System (CIFS) access to applications that support those standards. This will allow you to extend the reach of the DR550 archiving solution to a broader range of archiving applications

IBM also offers two tape classes, Linear Tape-Open (LTO) TS1030 and TS1040 and IBM 3592 TS1120. For each class, Read/Write cartridges and Write Once Read Many (WORM) cartridges are available. These can extend solutions such as SSAM and the DR550 by allowing older data to be migrated to tape while maintaining full WORM data protection.

3

Business drivers

IBM Information Infrastructure optimizes storage and content in term of its business relevancy to deliver solid, reliable information services. IBM Storage allows ECM customers to adapt to the requirements for their business without disrupting ECM applications by seamlessly growing as the business grows and making data available to the people who require it when they require it.

In this chapter, we examine business drivers and how ECM and storage, as the cornerstone of IBM Information Infrastructure, supports them.

3.1 IBM Information Infrastructure and ECM

The information age has transformed the free market economy, allowing traditional market advantages to be dwarfed by the advantage of information and the immediacy of its availability. IBM Information Infrastructure unlocks the value of digital assets by enabling organizations to create, harvest, and act on information in a collaborative manner. Enterprise Content Management and IBM Storage Solutions provide the foundation of an Information Infrastructure, by unifying unstructured data and making it actionable.

In today's marketplace, the ability to capitalize on information enables competitors to aggressively capture market share — and in a matter of months rather than years. The competitive landscape is such that long term prosperity is gained first by information empowerment, and second on the more traditional pillars of brand loyalty or market penetration. An organization's ability to unlock the value of its digital assets is quickly emerging as the standard on which corporations are differentiated.

The media wars of the early eighties illustrate how information is used to capture and dominate a market segment. Two formats, VHS and Betamax, emerged to satisfy the burgeoning appetite of consumers wishing to enjoy the Hollywood movie experience in the comfort of their homes. Although consumers would wrestle with the incompatibilities of these technologies for a decade, at its end JVC, the owner of the VHS technology, successfully diluted its competitor Sony and rendered Sony's Betamax format irrelevant.

JVC's product was not technologically superior to Betamax and had not been available to consumers for a full year after Betamax debuted, yet JVC still emerged as the dominant player. JVC's advantage was in their ability to leverage information and recognize market factors which indicated consumer availability was the key to winning the format war. JVC licensed the VHS technology to their competitors rather than trying to compete on its own, a strategy unpalatable to Sony. The competition between manufacturers drove down production costs and subsequently flooded the market with comparatively inexpensive VHS players. JVC won the war by capitalizing on information thereby attaining a superior understanding of the market.

In early 2008 Toshiba ceased competition with Sony over the emerging market for high definition movies, namely, HD-DVD and BluRay. Sony's understanding of the entertainment industry enabled them to successfully dilute their competitor. Instead of courting the consumer, Sony gained the commitment of movie studios to exclusively publish titles in Sony's format. The starvation of prospective titles allowed Sony to create the gap in market share to such an extent that Toshiba could no longer compete. While the video cassette wars spanned nearly a decade, the market for high definition movies was created and decided in less than a year.

Information empowers an organization to recognize and capture a market. The timely delivery of accurate and reliable information is critical in discovering and executing on those trends. IBM Information Infrastructure is the collective term used to describe the hardware and software that enable an organization to harvest and act upon the information they already have. The marriage of Enterprise Content Management with a robust and flexible storage infrastructure is key to maintaining the pace of the market and meeting the demands of the enterprise not only as requirements mandate today, but for the lifetime of the ECM solution.

IBM achieves this goal by making information as accessible and reliable as a utility not unlike electricity. Information is delivered with immediacy — resolute and unfaltering. Whether information is required to comply with audit requirements or is the fundamental differentiation between competitors, IBM Information Infrastructure is the platform for liberating digital assets and making them available on demand.

3.1.1 Leveraging digital assets for competitive advantage

Spreadsheets, rich text documents, and presentations are the foundation of employee productivity and the primary medium for creating of information. These assets, left unstructured, only benefit the audience with whom the asset has been shared. The value to adjunct business units is potentially unrealized due to the natural barriers imposed by departmental segregation and insular infrastructure design.

IBM Information Infrastructure liberates business assets and enables employees to base decisions upon a broader body of knowledge in a safe and secure manner. Such liberation is achieved by structuring data with business rules allowing LOB owners to organize, share, and retire information as business requirements mandate, and not leaving it to the discretion of the individual employee.

Enterprise Content Management, as the cornerstone of an Information Infrastructure, promotes collaboration and enables employees to execute on the strategic direction of the business. ECM also enables business owners to securely publish information appropriate for broader consumption. This capability can also allow partners and vendors to harvest the power of your ECM solution, enabling the consumption of information as a service, thus allowing cooperative partnerships to flourish unfettered by the barriers imposed by monolithic, people driven processes.

This unrestricted agility is complemented by a robust and flexible storage infrastructure. The complications of performance management and availability do not hinder the development of business processes, allowing business leadership to focus on the true objective — winning in an ever changing competitive marketplace. The fluidity of ECM is such that it can adapt to the evolving business model required to remain competitive, and IBM storage is the foundation on which ECM can remain fluid.

3.1.2 Ensuring resolute delivery of information services

A successful Information Infrastructure requires a robust and fluid storage architecture. As ECM enables the adoption of services and promotes collaboration, the storage infrastructure behind the solution must support the accelerated business activity and the constantly evolving manner the storage is utilized. Storage requirements for an ECM platform vary considerably from most other applications due to the broad capabilities of the platform. For example, storage requirements for defense contractors will vary considerably from the requirements of financial solution providers as each has differing use patterns and regulatory practices. A financial institution who trades securities might be required to store and retain records in a manner certified by the SEC, whereas a defense contractor maintaining information of national interest would require far more stringent rules.

As with differing industry segments, storage requirements also vary between business units within the same institution. A bank, for instance, has different requirements for mortgage processing compared to securities trading. Mortgage processing is a paper intensive workload due to the voluminous nature of modern real estate contracts and the number of external entities required to close escrow. As such, the life span of a typical mortgage might span forty or more years. Securities trading, on the other hand, is a very different case, because transactions are typically completed in a comparatively short period of time. Apart from government retention regulations, the relevance period of a stock trade is small compared to a mortgage transaction. As each LOB is responsible for maintaining differing retention policies, each has differing storage requirements.

Supporting ECM storage for multiple lines of business, each with a unique and evolving use pattern, necessitates a flexible storage architecture that can adapt on demand and with minimal disruption. An ECM solution, as with any environment driven by user generated content, places a continuously accelerating demand on storage. Ensuring that the storage infrastructure is utilized effectively is accomplished by determining by the value of the data relative to the cost required to maintain it. As storage demand grows, so does the collective value of the data, and in turn the potential financial risk imposed by the data being inaccessible. The storage underpinnings of an ECM solution must demonstrate unfettered availability, adapt to evolving business requirements without disrupting the accessibility of the data, and be accomplished at a cost appropriate for its value.

IBM Information Infrastructure optimizes storage and content in term of its business relevancy to deliver solid, reliable information services. IBM Storage allows ECM customers to adapt to the requirements for their business without disrupting ECM applications by seamlessly growing as the business grows and making data available to the people who require it when they require it. IBM ECM solutions allow enterprises to structure their data, tie content to reusable and renewable processes, and integrate with third party applications to effect business solutions driven by strategic vision and not application inflexibility.

3.1.3 Abating compliance and litigation costs

Unstructured data lives an indeterminate lifetime, is often difficult to locate, and might never realize its full business potential. Worse, it might realize its full liability as legal discovery rules could allow data stored on any medium to be actionable in a court of law. Such breadth allows prosecutors to mine otherwise benign data and present it out of context, contrary to the intentions of the creator.

IBM Information Infrastructure intelligently manages liability by segregating large cross sections of data at the record level. IBM ECM Storage further guarantees the record's integrity by preventing the record from being changed after it is committed. Additionally, many records do not have to be stored on costly near line disk. Those items relevant for legal retention purposes can be migrated to tape and stored off site, thus liberating datacenter managers from storing data whose business value does not justify the cost of monolithic disk. The cost of storing, cooling, maintaining, and powering disk is then available for investment in technologies furthering competitive advantage.

IBM ECM solutions are also able to automatically flag records that are eligible to be expunged. Configurable records management rules flag content that no longer has a quantifiable monetary value. The ability to purge data empowers business owners to hedge liability by limiting the proliferation of data no longer benefiting the business, while reducing the overall storage footprint at the same time.

Part 2

ECM storage reference architecture and products

In this part of the book, we introduce a Reference Architecture for the use of storage products with IBM ECM solutions. We then discuss both IBM ECM and IBM storage products.

4

Reference Architecture

In this chapter, we describe a Storage Reference Architecture for an ECM environment.

We discuss the following topics:

- Definition of a Reference Architecture
- Functional and non-functional requirements
- An ECM Storage Reference Architecture

4.1 The role of a Reference Architecture

A Reference Architecture is a blueprint for a solution in a specific domain. You will use a Reference Architecture to understand a solution's components, their interactions, and the reasons behind the choice of specific components. A Reference Architecture can also be used as a conceptual model that guides the choice of specific components.

The parts that make up a Reference Architecture are as follows:

- An overview of the various components
- A description of the relationships between the various components
- A description of the relationships between components that are part of the Reference Architecture and components outside of the scope of the Reference Architecture
- A description of the business requirements and how they are satisfied by the Reference Architecture

> **Note:** Components that are part of a specific Reference Architecture are those that are defined in the Reference Architecture itself; they are in the scope of the Reference Architecture. Components that are outside of the scope of the Reference Architecture, external components, are not documented in the specific Reference Architecture, because they will be the objects of a separate Reference Architecture.

A high-level Reference Architecture allows you to understand both the overall structure of the solution and the relationships between the various components, what we call the *functional elements*. Functional elements are elements that are instrumental to solving the business problem that is addressed by the Reference Architecture.

The Reference Architecture can be used as a guideline to outline and choose between the IT infrastructure components that can be used to implement the solution, and it allows you to understand the trade-offs involved in the choice of one solution component over another one. A classic example is the trade-off between performance and cost: the greater the performance required, the higher the cost will be for a specific component.

When choosing the IT infrastructure components that make up the Reference Architecture, we must also consider the so-called *non-functional requirements*. These are requirements that are not directly instrumental to the solution itself but are instrumental to the manageability and availability of the overall solution. Examples of non-functional requirements are:

- **Capacity:** This provides adequate resource to store the anticipated volumes of data.
- **Scalability**: This is a solution capable of growing to accommodate future demand.
- **Runtime availability**: The system must be as available as business demands dictate. High availability and business continuity fall under these requirements.
- **Systems management facilities**: The system must provide management interfaces such as user interfaces, reports, dashboards, and alerts to facilitate administration.
- **Security facilities**: The system must provide for secure data access and transfer, including the potential for data encryption.

4.2 Introducing the IBM ECM Storage Reference Architecture

We present a conceptual architecture for a storage solution targeted at an ECM environment. We discuss the ECM environment storage requirements and how these tie in to possible storage architectures that are available today on the market. We also emphasize that one storage size does not fit all and, as a consequence, we illustrate the necessity for processes, procedures, and tools to automatically manage the data during its lifecycle.

Enterprise Content Management (ECM) systems offer a diverse set of services to manage and store content for the complete duration of its lifecycle — that is, from the moment the object has been created, to the moment when it is deleted.

Consider the fact that an ECM system can also be used to give structure to unstructured data such as files or images. When files or images are stored directly on a file server, they are termed *unstructured data* because there is no way to correlate between different files or images, whereas when files or images are stored in an ECM application, one can correlate images and files to specific business processes.

We can draw a conceptual representation of an ECM system identifying four different levels of components in the overall architecture. These levels, illustrated in Figure 4-1, are:

- **Clients:** Users and applications that access the data in the ECM system
- **Services:** Implement business function and facilitate access and use of the data
- **Repositories:** Containers that store the data and metadata associated with the data
- **Storage infrastructure:** Can represent both physical hardware devices that store data and metadata, as well as software components that offer storage and storage management functions

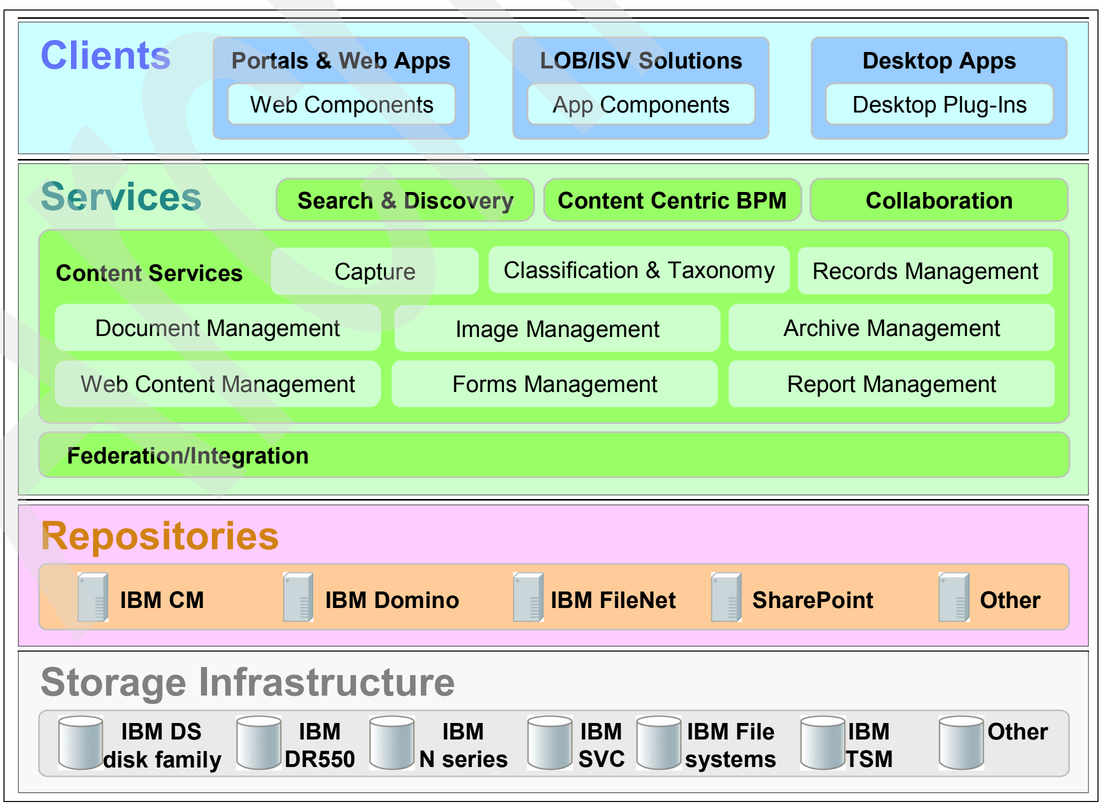


Figure 4-1 Functional layers in an ECM solution

We describe all four levels in detail and then go on to analyze further the role and interactions of the three lower levels: services, repositories, and storage.

Storage Infrastructure: At the bottom of the stack we have storage infrastructure components. These are the components that allow you to store the data. Each individual component can also offer additional storage related functionality such as data replication to a standby storage device, data migration to cheaper storage devices, or WORM type features so that the data cannot be modified. In the storage infrastructure layer, we include both hardware storage devices and storage software implementations such as file systems and storage management products.

Repositories: Next to the bottom of the stack, you see a number of different repositories, or places where content and data are stored. Information can be stored anywhere – in IBM repositories, non-IBM repositories, file shares and e-mail systems, databases, and storage devices. A typical company might have multiple repositories such as file servers, e-mail systems, engineering, and financial systems. There is a requirement to be able to access data across all these repositories.

Services: Above the repository layer, you can see a set of services that provide active management and utilization of the information. These are the core capabilities that manage all information types, whatever type they might be. They could be images, reports, documents, e-mail, Web content, forms, audio, video, or XML.

To connect to all the repositories where information resides, ECM offers information integration and content federation services. This is more than just a connection to one content repository or another. It provides active management of content “in place,” so you do not have to migrate or move content. It helps you retain, find, use, reuse, and manage information regardless of where it resides.

ECM also offers a complete set of information management capabilities to bring life to the content, from capture and management to presentation, search and archival, and everything in between. ECM solutions also offer compliance and records management solutions.

Clients: And finally, on the top of the stack, are the client interfaces. These client interfaces are accessed by end users and are a gateway to reach the information stored in the content management system.

Each of these elements are critical for Enterprise Content Management. From the storage perspective, the areas that are most relevant are services, repositories, and the underlying interaction with the storage infrastructure.

The choice of the underlying storage solution depends on the business requirements of the ECM application and how they map to storage requirements. In the following sections, we discuss these considerations.

4.3 Functional requirements

Here we discuss the functional requirements regarding the use of storage by ECM applications. These functional requirements will guide and drive the choice between the various storage devices that an ECM application can access and use.

We can break down these functional requirements into four main groups of attributes.

- **Information Security:** Related to data protection, encryption and sharing and avoidance of unauthorized access
- **Information Integrity:** Related to risk and data integrity, protected or unmodifiable storage
- **Information Retention:** Placement and movement of data between the most cost-effective storage tiers and timely deletion of old data
- **Information Availability:** Related to high or continuous availability and the access performance to data

Last but not least, we must consider the solution's total cost of ownership (TCO) as all of the storage services required to satisfy the requirements come at a cost. Next, we discuss each of these four groups of requirements and TCO aspects in more detail.

4.3.1 Information security

The first functional requirement area is information security and the underlying data security. What kind of security does the underlying data require? One must ask if the ECM application will be used to store sensitive information, such as information that could cause harm if it falls in the wrong hands. Such information could include medical data, customer records, personal data, and trade secrets.

So long as the data is stored in a secure data center, there are no particular exposures in the use of standard storage devices. If the data has to be sent offsite for business continuity or disaster recovery, then consider encryption. Encryption can be implemented in various storage components. High end tape storage devices offer the ability to encrypt the data on tape so it cannot be accessed even if the tape falls into wrong hands. There are other implementations of encryption you can consider, such as Tivoli Storage Manager and DR550 encryption or network encryption of data before sending it over a WAN link to a remote site.

4.3.2 Information integrity

The second functional requirement area is that of minimizing the risk to reputation and of failing an audit. One key aspect of integrity is avoiding the accidental or malicious modification or deletion of data. Can an administrator or an end user be allowed to modify data either intentionally or unintentionally? What are the potential consequences of such a data modification?

There are various IT enablers that can address information integrity requirements:

- **Protected storage:** Storage devices that do not allow data modification or deletion before a predefined expiration date
- **Data Governance policy automation:** Clear and well defined IT processes and procedures that define the data lifecycle and policy automation tools to automate the business processes

From the storage perspective, protected storage devices — that is, devices that do not allow data deletion before a predefined expiration date — often have a key role in the solution.

Information integrity is also offered by devices that allow data protection. This protection could be implemented with local RAID type protection and also remote replication for decreasing the risk of data loss.

4.3.3 Information retention

Information retention covers aspects regarding the conservation of data for the amount of time required by the application as dictated by the business rules or legal requirements in the most cost-effective manner.

The *value* of information can vary over time. For example, with stored customer transaction statements, the probability of use is higher when the statement has just been received by the customer, whereas the probability of use is much lower for a statement that is one or two years old. The consequence of this is that you could decide to store current statements on storage devices with higher performance and availability characteristics, and decide to move older statements to less expensive storage devices. This would allow you to streamline the overall cost of the underlying storage infrastructure.

Some examples of IT enablers that address information retention are:

- **Tiered storage:** The use of multiple levels of storage devices based on the cost per unit of storage. Cost is the most important driver for multiple storage tiers.
- **Policy based archiving:** Storing data on storage media and moving data between different storage media tiers according to predefined policies. Archiving is often implemented by using software products.
- **Event based data retention:** Addresses the retention and deletion of data based on specific events. For example, determine when a life insurance policy must be paid back.

Information retention covers aspects relating both to the retention and to the deletion of the data when business rules dictate that is no longer required. Information retention IT enablers such as archiving and tiered storage are motivated primarily by IT infrastructure costs.

4.3.4 Information availability

Information availability is key to the smooth running of business processes. The basic requirement is to continuously and reliably deliver information where it is required, when it is required. A disruption in data availability can have a noticeable business impact. Beyond the lost productivity costs, it could cause customer dissatisfaction and even the loss of new business.

Data availability also comes into play when you have to deliver a predefined service level to the business. Data infrastructure redundancy helps guarantee data availability in cases when one component of the data infrastructure is unavailable.

There are multiple storage solution enablers for data availability. Here are some examples:

- Advanced data mirroring functions can be implemented in multiple storage devices to replicate a copy of data to a separate remote storage system so that it can be accessed in case the primary storage system fails.
- Data consistency enablers can guarantee that the data and the ECM system metadata are in sync at all times, even in the case of individual component failure.
- Virtualized storage infrastructures can facilitate scalability and operational tasks, such as component migration and replacement.
- High performance storage devices can allow you to guarantee application response times even under heavy load conditions. High performance storage might also be required for non-functional requirements such as data protection with backup products.

Other important aspects to consider are the Recovery Time Objective (RTO), or time to reestablish service; and Recovery Point Objective (RPO), or amount of data lost for the application, and thus the requirements for the underlying storage devices. These terms are often used when discussing disaster recovery, but also apply to local recovery requirements.

- RTO defines the time required to recover the application in case of component or site failure. Starting from a mirror copy on a separate disk device is often faster than having to recover all the data from tape.
- RPO defines the currency of the data used to restart the application. When recovering data from a daily backup, the RPO will be about one day, whereas when recovering from replicated disk, the data will be much more current.

RTO and RPO requirements are important drivers in the choice of storage solutions.

4.3.5 Controlling IT storage infrastructure costs

All the foregoing requirements for data and information must be mapped to the budget for the ECM and storage IT infrastructure.

The total cost of ownership of an IT infrastructure can be broken down into two cost metrics: capital expenditure (CAPEX) and operational expenditure (OPEX):

- CAPEX is the initial, up front, capital outlay to acquire, set up, and integrate an IT component into the overall IT environment. CAPEX includes acquisition, installation, and configuration costs, as well as costs related to the asset's integration into the current business processes.
- OPEX accounts for all the ongoing recurring operating costs to manage and run the IT infrastructure in question. OPEX includes system management, operator, maintenance, data center power, cooling, and supplies costs.

In general, ECM systems require storage of content for long periods of time, from years to decades — periods that might easily outlive the useful life of individual hardware components. For this reason, it is important to clearly understand the operational expenditures of an IT infrastructure. Aspects such as power and cooling become important: High performance disks usually consume more power than low performance disk devices; tape devices consume much less power than disk devices. So moving data between storage tiers can be cost effective.

A second important aspect to consider when evaluating storage infrastructures for ECM systems is the possibility of moving to different storage devices as these become available. Carefully evaluate whether there are migration paths to replace one storage device with another at some time in the future. This is the key to infrastructure flexibility.

4.3.6 Data use and reference patterns

Some important aspects to consider when defining and designing an ECM storage solution are data use and reference patterns —the way data is used by the application. These aspects include retention requirements for specific data types and access frequency to the data.

Next, we illustrate some data access and use attributes that you have to understand so as to define the optimal storage solution:

- **Retention requirements:** Does the data have to be retained for a defined period of time for corporate governance or regulatory requirement purposes?

- **Disposition requirements:** When data has reached the end of its retention requirement, can it be disposed of completely (in other words, deleted), or must it be moved to lower cost media?
- **Archival requirements:** Define whether the data has to be kept for long periods of time. Does a copy of the data have to be archived and stored separately from the original data?
- **Access frequency:** What is the frequency or probability of access to the data after it has been created? How frequently or infrequently is the information accessed once created? Will the data be accessed often, rarely, or probably never?
- **Read and write performance cycle:** Considering the access frequency and probability, what are the performance requirements for both read and write operations? What are the appropriate and most cost-effective technologies to satisfy these requirements?
- **Read and write:** Does the data have to be stored on non-erasable, non-rewritable media, that is, “write-once-read-many” (WORM)?
- **Data access times:** What are the desired speed and response time to recover the data?

These are only some examples of criteria and requirements to keep in mind when defining the underlying storage solution.

Having defined a set of functional requirements, we now consider what other qualities or non-functional requirements our architecture must be manageable and resilient enough to satisfy the required service levels.

4.4 Non-functional requirements

Non-functional requirements define qualities and constraints that the IT infrastructure must satisfy — qualities and constraints that do not relate to the primary functional requirements. As an example, functional requirements for a disk storage system could include capacity, performance, and cost. Non-functional requirements could comprise ease of installation and configuration, power consumption, security aspects, and so on. We choose a specific IT component based on the functional requirements and verify that it also satisfies the non-functional requirements.

4.4.1 Availability

The availability requirements of a system and the underlying infrastructures are defined by the hours of uptime required in a given period. Availability also defines how fast applications must be recovered from a failure. Availability requirements will span the servers that run the application, the storage devices that contain the data, and the networking equipment that the servers use to access the storage devices.

Often the expected availability of a system is 24 hours a day and 7 days a week (24x7). To attain such high availability, the costs can rapidly escalate to levels not warranted by business requirements. When defining availability requirements, it is important to plan for maintenance windows. For example, you could plan for a weekly maintenance window of one hour for scheduled maintenance. The system will not necessarily be shut down each week for the defined period of time, but if necessary, the maintenance window is defined and stipulated in the set of non-functional requirements.

A sample availability non-functional requirement could be stated as follows:

- The IT infrastructure component availability is 24x7 with a downtime allowance of two hours a week for planned maintenance operations.

- Recovery from any unplanned IT infrastructure outage will be successfully completed within one day of the outage.

4.4.2 Backup and recovery

To ensure that the data is available and can be recovered after a system failure or outage, you can define backup and recovery requirements, which must be defined in accordance with the availability Service Level Agreement (SLA). Recovery definitions and the type of data definitions must be clearly understood when choosing the underlying storage technology and defining the backup and recovery processes. Recovery definitions define the requirements in terms of RTO and RPO, so they dictate the type of backup solution, which could be data mirroring or a periodic backup and restore operation.

It is also important to understand the requirements of the different types of data, or data categories, such as these:

- **ECM database data:** This kind of data probably requires a high level of protection. Database transaction logging must be enabled to allow for database roll-forward recovery. We might have a daily full backup of the database and periodic archiving of the database logfiles.
- **ECM content data:** The data on file systems will be backed up incrementally on a daily basis. A monthly image backup operation will be performed for fast restores of the whole file system.
- **Static data:** This category includes program files and operating system data. This data will be backed up on a daily basis.

4.4.3 Business continuity and disaster recovery

Business continuity and disaster recovery requirements define the ability to adapt and respond to risks that can affect the IT infrastructure. The IT solution must be resilient enough to guarantee the application and IT infrastructure SLAs. Multiple non-functional requirements are closely tied to this non-functional requirement, for example:

- Availability
- Backup and recovery
- Disaster recovery
- Scalability
- Capacity

Business continuity in itself can be implemented with redundancy. Redundancy at the component level is well understood, it covers aspects such dual paths to data, RAID protection schemes in disk storage devices, and clustered server for local application servers.

Disaster recovery refers to those situations where the whole of the datacenter, all redundant components, have been lost. If a disaster such as fire, flooding, or earthquake hits the primary IT complex where the systems are hosted, you must have a plan defined and in place to recover business applications in a reasonable timeframe. If your IT operation already has a disaster recovery (DR) plan, the ECM application and the underlying storage infrastructure have to be integrated in the current DR plan. The DR plan is an IT-focused plan whose scope is to restore the operations of the target systems and applications at an alternative site after a disaster has hit the primary site.

In most cases, DR planning includes an analysis of your business processes and their continuity requirements; you must also focus on disaster prevention. Likewise, consider the disaster tolerance, that is, the IT infrastructure's ability to withstand major disruptions to the underlying systems. Disaster tolerance has to be built into an environment at various levels. This can take form of hardware redundancy, high availability/clustering solutions, multiple data centers, eliminating single points of failure, and distance solutions.

4.4.4 Capacity planning

The architecture of the IT infrastructure being designed must support not only the current load but also the projected load on the system over a reasonable period such as 3-5 years. Load on the IT infrastructure is not just a factor of the number of users accessing the system concurrently. Rather, it is usually based on a combination of a variety of system usage characteristics and attributes, for example, concurrent background housekeeping tasks such as data backup. The IT infrastructure must be designed to handle both average loads and also give adequate response time during peak load periods that might recur cyclically at certain times of day or year.

Metrics to consider are the amount of data objects inserted into the system — the ingestion rate. Do data objects arrive gradually over time or are they loaded onto the ECM application with batch processes at predefined times? Can objects be stored in a staging area before being stored in their final destination.

4.4.5 Extensibility and flexibility

Extensibility and flexibility is the ease of extending the IT infrastructure architecture to accommodate new business functions and include new technologies in the future. Business requirements change over time because the business grows and evolves, so the Reference Architecture must be extensible to accommodate future changes. It must also be able to accommodate extension and change without too much disruption or reconfiguration so as not to impact the currently running application.

The way to accommodate leading edge technology and innovation must be taken into account because there are high chances that the ECM application will have to be around for years or decades.

4.4.6 Performance

It is not easy to define a response time for a system as these tend to be quite subjective and to differ between different kinds of users and different applications. Response time performance characteristics must be defined based on the real business requirements.

The overall response time represents the end user response time, the time from the start of a request to the return of the response to the user. The final user response time can depend on elements outside the scope of the Reference Architecture, for example, the response time might also depend on the network that is used.

Example response time characteristics could be stated as follows:

- 97% of all inquiry transactions must complete within 3 seconds.
- 97% of update transactions must complete within 5 seconds.
- The remaining 3% of transactions must complete within 120 seconds.

This exception case takes into account that there are probably some *heavy* transactions that require a large amount of data to be retrieved or updated. It could also include infrequent accesses to very old data that has been migrated to slower, cheaper media.

The last criteria is especially useful when designing ECM repositories that include tape in the storage tiers. Time to first byte of data on a tape device is often around 60-90 seconds.

The system or ECM application must be able to measure and record performance metrics at a system component level so that appropriate reports can be generated for monitoring purposes.

4.4.7 Reliability

Reliability is the ability of an IT infrastructure to deliver application services in a predictable and consistent manner, both under normal circumstances and under unexpected conditions. Here are some requirements for a reliable IT infrastructure:

- Return the same data and results under any type of system condition: both normal load and high stress conditions.
- A transaction failure must not result in non-recoverable data loss or data corruption
- In case of failure, the IT infrastructure must report clear error messages to the user with suggested recovery steps.

4.4.8 Scalability

Scalability is defined as the ability to expand or even contract the IT infrastructure to accommodate changes in business requirements such as additional load, users, and data. A scalable IT infrastructure is extendable without requiring a complete replacement. This non-functional requirement will affect both the Reference Architecture and also the underlying hardware systems and software components.

The infrastructure must support scalability across three separate domains:

- Hardware
- Software
- Applications

In each of these domains, the architecture must provide both horizontal or scale out scalability and vertical or scale up scalability.

In IT infrastructure terms, horizontal scalability or scale out involves adding more components such as more servers or storage devices, whereas vertical scalability or scale up involves adding more capacity to existing components, such as adding more processors or memory to a server.

To understand how much the system has to scale, you must consider current load and anticipated loads over the 3 to 5 year time frame under consideration as well as functionality. Scaling does not have to require significant software, application, or database changes. The IT Infrastructure must be capable of providing for unpredictable growth such as peak seasonal demands.

4.4.9 Security

The Reference Architecture for the IT infrastructure must consider the security requirements for the application. Consider both physical and logical security aspects and give thought to the following aspects:

- **User authentication and identification:** To protect against fraudulent access to data and information and avoid deception by using the information fraudulently obtained
- **Access control:** To prevent unauthorized use of a resource
- **Integrity:** To ensure the correctness of information and data stored in the system and returned by the application
- **Accountability:** To ensures that the actions of a user can be traced back to the user
- **Physical security:** Comprises both access security to the datacenter and methods such as data encryption to ensure that data sent out of the datacenter cannot be used if it falls in the wrong hands

4.4.10 Service level agreements

Many of the non-functional requirements are both directly and indirectly instrumental to defining service level agreements (SLAs) for the IT infrastructure. Some examples of typical aspects to include in SLAs could be:

- **Availability:** The system will be available 24x7 with an allowable down time of at most two hours per week.
- **Response time:** The system will give response times in the order of xx seconds for 97% of all transactions.
- **Measurement:** System statistics, transaction numbers, and response time statistics will be recorded, and current and historical reports will be provided on a weekly basis.
- **Backup and disaster recovery:** Backup will be performed daily and a copy of the data will be stored on tape and sent to an offsite vault.

4.4.11 Standards

IT architectures have to be built upon industry standards, because doing so will facilitate co-existence and interchangeability of multiple technologies and tools. Complying with standards will give you more choices and flexibility to react to changes in business or technology conditions without restricting the overall solution architecture. Consider the following design criteria when evaluating the standards:

- Is the standard being evaluated available now on the market and can it be implemented today?
- Does the standard agree with the high-level architecture? The standard must not add more overhead, in which case you have to consider not using the standard.

4.4.12 System management

Systems management requirements are an important aspect of non-functional requirements. A Reference Architecture must include the definition of the management processes for the components that constitute the reference IT architecture. There are various general architecture guidelines to consider when defining component management characteristics.

The architecture must provide sufficient management services to enable you to satisfy the service level agreements. To achieve this, parts of the architecture must be managed as stand alone entities, but all parts must participate and be integrated in an overall systems management architecture. Next we describe some of the areas of requirements for the architecture and management framework.

Command and control: At a minimum, services to start and stop individual components must be provided. Then it has to offer services for load balancing to spread workload over multiple components and physical servers, and as an extension to this, the architecture must also offer failover capabilities.

Local, remote, and programmable configuration: All components must be capable of local and remote configuration and allow for an automated scripted setup.

Logging: The architecture must provide standardized services for general logging of events and failure conditions.

Event notification: The architecture must provide standardized services for generic event notification.

Monitoring: It is up to the enterprise management framework to provide the overall monitoring of the environment. The architecture must work closely with the enterprise management framework to forward the appropriate event notifications.

Performance: The architecture will be able to record and monitor the performance and utilization of individual resources in the system. The architecture has to support the extraction and analysis of this information.

The architectural system management requirements illustrated above are a basic set of requirements that must be satisfied in any Reference Architecture.

4.5 The IBM ECM Storage Reference Architecture

We now illustrate a Storage Reference Architecture for an ECM environment. An ECM system is architected to store data and perform four basic functions on the data objects that it receives from user or application interfaces and stores on the underlying storage devices:

- **Create:** Create new data objects and store them on the most appropriate storage device.
- **Read:** Locate and retrieve one or more specific data objects from the storage repository.
- **Update:** Locate and retrieve an object that has been stored, change it, and store it again.
- **Delete:** Locate an object and delete it from the ECM system. The object might or might not be deleted from the underlying storage device.

Our ECM Storage Reference Architecture's aim is to satisfy these four basic functions in the most efficient and effective manner.

We start by presenting the various components or layers and discuss the functions that each layer performs and how each layer interacts with the other layers. Figure 4-2 illustrates the IT services and components of our functional architecture.

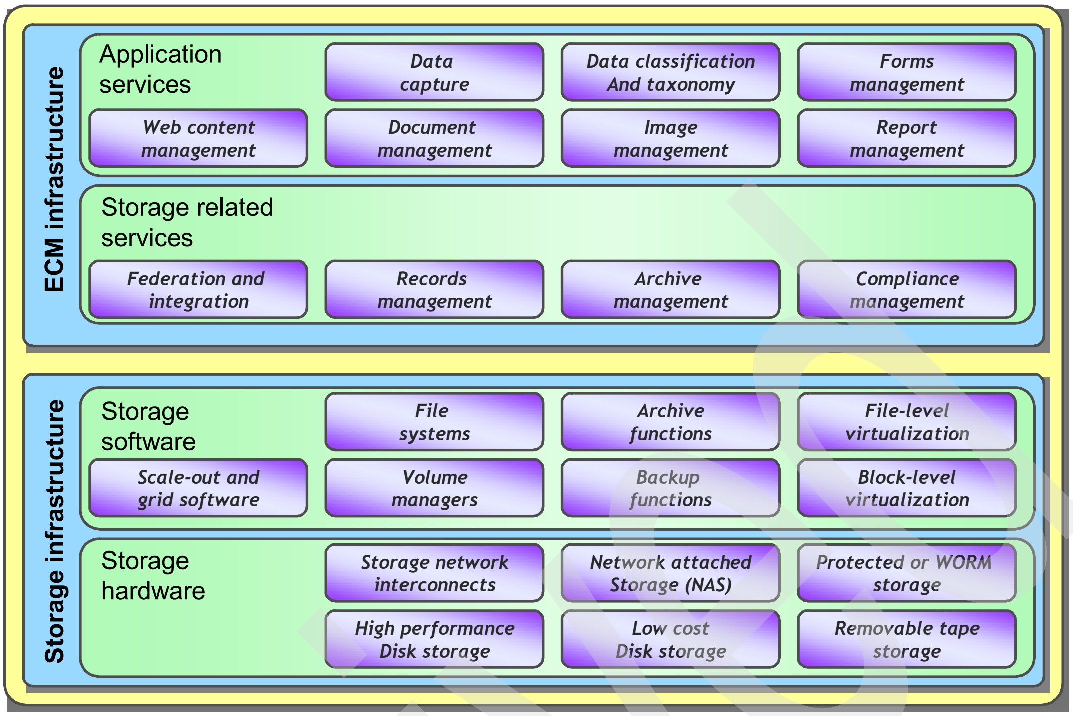


Figure 4-2 ECM architecture functional components

We have separated the components into two separate domains that relate to where these services are provided. These are, respectively, the ECM infrastructure domain that contains all the services offered by the ECM infrastructure, and the storage infrastructure domain that relates to storage services and technologies.

Note that there might be some level of overlap between the two domains. One example of this overlap is archive management, moving data to a different storage media. The function could be implemented either in the ECM domain and performed by the archive management service or in the storage domain by file-level virtualization or archive management functions. As a general rule, the closer the service is performed to an application, the better, as the application will have greater control of what data to manage.

4.5.1 Storage related ECM services

We have the *ECM infrastructure* domain, which contains functions and services that are usually part of an ECM product. The *application services* are those that are not directly storage related, whereas ECM infrastructure *storage related services* are those that control and manage the use of storage. We have identified four storage related services and processes in this area.

Compliance management

This is more of a business process and not so much a service. Compliance management is functionally the layer at which business rules and business controls are defined. Both regulatory and business requirements for the ECM application are evaluated and documented. Based on these requirements, data classification rules can be defined, which are rules that define how each individual kind of data has to be managed, what the high level storage and retention requirements must be, and who has access, control, and responsibility for the data. Data classification rules are often called the *file plan*.

A file plan is the classification of data according to the business rules, essentially a set of rules that allow appropriate storage and management of data. A file plan is the output of the compliance management process and input to the records management service.

Records management

This is an IT function or service that allows for implementation of the business file plan. Records management is all about control; it makes sure that you only keep what you have to keep for as long as you have to keep it and then afterwards ensure that it is securely destroyed. The file plan that is the output of compliance management is the input to records management. Records management is the process of implementing the rules and guidelines defined in compliance management and that are documented in the file plan. The records management service can be a separate add-on product or can be integrated into the content management product itself.

To clarify the term *record*, it is an object than contains both metadata attributes such as user, date, and business application, and data such as multiple images and documents. So a record can contain a collection of different objects. Records management allows specific objects to be stored in selected storage repositories, for example you could choose to send text documents to high performance disk and images to lower cost storage devices.

Records management starts with *records declaration*, that is the way to call or interface with a records manager. Automated or touchless records declaration could be implemented as a step in a workflow process, a scheduled batch job, or a document lifecycle transition, whereas manual records declaration implies an action by the application owner such as using drop-down menus. Once declared, you have records that can be searched and viewed based on the security rights that the user has.

When the record has been declared as such, the record is passed to *records classification* where the business rules defined in the file plan are applied to the record itself. Records classification could be either automated, based on record metadata, or manual, where the user selects the required classification. Records classification applies formal, rules-based management to the retention and disposition of that stored content. Records classification rules define the storage repository destination of the records data; these repositories are then managed by the archive management function.

Records management can also alter the records retention, for example the retention period for a record might have to be extended due to new business requirements or an external request. In this case a deletion hold can be placed on the record to avoid the scheduled deletion of the record.

Archive management

Archive management is the service that is responsible for storing the data onto the underlying storage infrastructure. The records management service defines where record data must be stored. The archive management service performs storage of the data in predefined storage containers or repositories, as well as the management of the stored data for its complete lifecycle, from creation to disposition. There are different phases in the lifecycle of record data:

- **Creation:** The moment when new data is received by the archive management service. At this time, the archive management service decides the appropriate repository in which to store the data.
- **Movement:** The archive management service can move data between storage repositories and tiers based on business rules and events.
- **Disposition:** When the data reaches end of life, it will be deleted from the storage device, or it could also be archived to lower cost media to save on primary storage costs.

Archive management can store data in multiple storage repositories. You could store the same data into two different repositories for data redundancy, or store different kinds of data in different repositories based on data access characteristics.

Archive management must strive to make the most efficient use of storage repositories, for example, by balancing or spreading out new storage allocations to separate devices to optimize performance and overall capacity utilization. The archive management function must also offer data movement functionality to move records to different storage tiers based on specific business rules. Data movement functionality must be available both by date and by event, so that individual records or groups of records can be moved based on business rules.

The archive management function must also offer instrumentation to report on current and historical storage utilization trends and data access and reference patterns.

Federation and integration

These services allow you to seamlessly access and aggregate data across multiple storage repositories, repositories that could also be implemented using different products. Data federation is the ability to transparently access diverse business data from a variety of sources and platforms as though it were a single resource. A federated data server can access data directly, for example, by accessing a relational database, or it can access an application that creates and returns data dynamically, such as a Web service.

The federated approach to information integration provides the ability to synchronize data without having to move it to a central repository. Data federation is a way to hide the complexity of accessing multiple data stores. Data and content federation can simplify complex information access requests by unifying information across databases, data managers, data models, and technology platforms.

Information integration builds upon the foundations offered by information federation services. Information integration provides a comprehensive solution for simplifying the management of the great volumes and diversity of data that exists in enterprises and organizations. More and more IT operations involve the requirement to integrate diverse and unconnected infrastructures and platforms.

Federation services can also be used as a tool to move or migrate data between different repositories.

4.5.2 Storage infrastructure software services

There is not a clear distinction between storage infrastructure software and hardware layers. Because the same function can be implemented at different levels, it is possible to have the same function both in software and in hardware. There will be cases where there is an overlap of function offered in storage software or hardware, or even between ECM infrastructure and storage infrastructure.

File systems

A file system is a layer of abstraction, a layer of virtualization, that simplifies the access to data. A file system consists of a set of abstract data types that facilitate the storage, retrieval and management of files. A file is a set of information, usually stored on durable storage devices and referenced by name — the filename. Often files are named based on the content they store and, on some operating systems, an extension is used to associate the program required to access that particular file. So a file system allows you to name data according to its content and later store it and address or retrieve it by the content’s name.

The use of file names frees the user from the necessity to remember the physical location of data on the storage device. The user can access the file via its name and the file system will locate and return the data; in this way, the user is insulated from knowing or managing the underlying semantics.

File systems also can provide additional storage management services. Files can be moved or copied to different destinations in order to optimize the storage of files when they become fragmented, or scattered over the physical storage medium. File systems offer security mechanisms so that individual users can access only the subset of files to which they have been granted access. File systems, such as IBM AIX® JFS2, offer snapshot functionality to create fast logical copies of data.

There are many different kinds and implementations of file systems. The two main categories are local file systems and network file systems.

- **Local file systems** are usually implemented in the operating system or as add-on software products to the operating system. The majority of current operating systems do supply their own file system, for example, WIndows NTFS and IBM AIX JFS and JFS2, or you can acquire specialized file systems. The local file system is an integral part of the operating system and serves files only to applications running in that specific operating system. You cannot access a file on a different computer through a local file system. You require a network file system.
- **Network file systems**, as the name implies, are accessed over a network connection and permit multiple computers, often referred to as clients, to access and store data on a central computer commonly called the file server. Common examples of network file systems are the NFS and SMB or CIFS protocols. A network file system allows multiple users to share files and often will provide a locking mechanism to prevent multiple users from updating the same file at the same time.

File system performance

File system performance depends on many aspects. First, let us define two performance indicators:

- The time to locate a particular file, given its name, and start accessing the data
- The time required to list all the files in the file system, also known as traversing the file system

Both performance indicators are impacted by the underlying storage hardware characteristics such as response time and sequential throughput. Response time and traversal time are also a function of the number of files in the file system. The larger the file system, the longer the time required to traverse it and enumerate all the files.

In an ECM implementation, carefully consider the type of file system you want to use, based on the number of files that you expect to store in it and the type of access to it. Storing too many files in the wrong type of file system can lead to severe performance degradation. This will probably be felt more during housekeeping procedures such as periodical backups of data, where all of the file system will be traversed to determine which files to back up.

Volume managers

In contract to a file system, which presents and manages files, a volume manager, also termed a logical volume manager (LVM), is used to access and manage underlying storage volumes, often referred to as physical volumes (PVs) or LUNs. LVMs simplify the allocation of mass storage devices and management operations. An LVM can be used aggregate multiple physical LUNs into a larger partitions using techniques such as concatenation of physical LUNs or striping the data across multiple LUNs. Major LVMs will allow you to concurrently move or resize a a partition without impacting application availability. Some volume managers offer snapshot functionality to create copies of data. Volume managers, in the same way as file systems, are often implemented in the operating system or as software add-on products to the operating system.

In an ECM environment, as in other environments, the use of a volume manager can facilitate data management operations such as adding capacity or migrating to new storage hardware.

Archive and HSM functions

Archiving and hierarchical storage management (HSM) are functions that are used to optimize storage utilization and contain costs of storing large volumes of data. In general, archive and HSM techniques are applied to files that reside on a file system. Archiving and HSM are functions that allow you to move less important data to less expensive storage tiers and thus control the cost of storing large amounts of data:

- **Archive:** This is commonly identified as the process of selecting a group of files according to some common criteria and moving them to a different, usually less expensive, storage tier. The concept of moving implies that the files are copied to the new tier and then deleted from the original file system. The archive function is invoked from a higher layer such as an ECM application service. The application using the archive function must keep track of where the files have been archived to, because the file system no longer contains any information on them.
- **HSM:** This is a technique to automatically move data between storage tiers based on predefined rules. HSM allows you to store the bulk of your data on low cost storage devices and move it to higher performance storage devices when it is required. HSM is usually implemented at the file system layer as an extension to the file system. Files are migrated, that is copied and deleted, to lower cost storage tiers either by command or, more often, by automated processes that evaluate attributes such as last use date and size to decide which files to migrate. Migration processes are usually automated and start when the file system exceeds a predefined utilization threshold.

  HSM leaves a trace or marker in the file system, sometimes referred to as a stub file, so that in the user's perspective, the file still appears to be there. When a migrated file is referenced (for example, opening a Microsoft® Word document), the HSM service is transparently invoked, and performs one of two actions; it either recalls the file to the original storage tier or redirects application access to the migrated file on the new storage tier. Unlike in the archive case, the application does not have to be aware of files that have been moved to a different storage tier as this is tracked and handled by the HSM function.

  HSM is a good choice for ECM applications if they store medium to large files and do not want to handle the underlying storage management tasks. HSM tends to be less efficient with small files, for example under one MByte in size, as the space savings are not very great and the HSM overhead can start to be significant depending on the implementation.

Archive and HSM implementations can be multi-tier, as the availability of low cost SATA disks has created a market for three-tiered solutions:

- **Tier 1:** High performance Fibre Channel storage devices
- **Tier 2:** Lower cost storage random access devices such as serial ATA disk
- **Tier 3:** Lower cost tape storage devices

Tape can be part of the storage tiers for infrequently accessed data, due to the fact that with tape, it takes time to mount and position the cartridge before data can be returned to the application. These times are in the order of 60-120 seconds.

Backup function

The term backup refers to taking copies of data and storing them on separate media so that the original data can be recovered in case of primary data loss due to logical data corruption or physical storage media failure. These copies of the data are called backups. Backups are the last line of defense against complete loss of data. Do not confuse backups with archives. Backups are a second copy of data, whereas archives are the primary data itself. If you lose a backup and still have the primary data, it is not a problem, whereas if you lose the archived data and there is no backup, then you cannot retrieve the data.

Backup storage requirements can be considerable because they imply at least one copy, (often more), of data to be stored on backup media. Because of this, backups are usually directed to cost-effective media such as tape. Tape is also an optimal solution because backup data is used infrequently and usually in case of an unplanned event such as accidental data loss.

The major enterprise backup software implementations consist of a catalog database that inventories the data being backed up and one or more storage repositories to store the backups themselves. Plan to have the catalog database on disk storage devices with adequate performance. The choice of storage repositories is more varied; disk and tape devices can be used. A good backup product must also allow you to stage data on disk before sending it to tape, this is sometimes referred to as disk to disk to tape (D2D2T). Staging backups to disk will allow you to perform multiple concurrent backups, al LAN speeds, to the same disk storage device. The data can later be destaged to tape devices at locally attached device or SAN speeds.

Backups must be stored in multiple copies; enterprise backup products will allow you to create multiple copies of the same set of backup data. It is good practice to keep one copy of backup data onsite for recovery purposes and a second copy in a remote vault location for disaster recovery situations.

The process of sending data to a backup storage device implies various processing steps to optimize overall backup performance. These steps might involve ways to handle open files, data from databases, and other applications, space saving techniques such as compression and deduplication, as well as encryption.

Major backup software products allow you to set up policies defining versioning and retention of data, the destination where to store it, and the scheduled frequency of the backup operation. They offer backup techniques such as full and incremental backups, file system backups, or the protection of a complete volume or partition using raw partition dump techniques.

The backup of ECM systems requires that ECM data and ECM metadata must be backed up at the same time to guarantee consistency. One way of doing this is to close the ECM application for the duration of the backup operation.

ECM data tends to be relatively static, that is the arrival and change rates are low compared to the volume of stored data. This implies that daily incremental backups will have to traverse a relatively large file system and to back up a small quantity of data. If you must restore the data, you might have to restore it from multiple incremental backups. We suggest that you consider periodically performing a full or raw file system backup in order to speed up restores.

Archived ECM data must also be considered in the backup strategy. If old or seldom used data is archived to a low cost storage device such as optical disk or tape, this data must be protected with a backup solution. We suggested that the archive product used must be integrated with the backup product.

File-level virtualization

In the computer world, virtualization is a technique that refers to hiding the real physical characteristics and interfaces of physical computing resources. It also refers to making one or more physical resources available as one or more logical resources. One physical resource to many logical resources is often called *partitioning*, whereas one logical resource to multiple physical resources is often termed *resource aggregation*.

File-level virtualization software can offer services such as accessing and storing data in multiple physical repositories as if they were one. File level virtualization can also be used to move and migrate data between different repositories or to replicate data between two different repositories. It is called file-level virtualization because it acts on files and usually operates at the file system layer.

File systems offer a certain level of file-level virtualization because you can seamlessly mount and access data using a common directory tree name space. UNIX® offers the **`mount`** command to connect and access one file system as a subdirectory inside another file system. Current Windows® operating systems offer similar function.

File-level replication is another interesting aspect that can be leveraged in ECM solutions to synchronize data between different storage devices.

Some NAS storage products can offer file level virtualization by aggregating multiple diverse NAS devices and making them appear as one unique entity and one namespace.

Block-level virtualization

Block-level virtualization is similar conceptually to file-level virtualization, with the difference that file-level virtualization operates on files, whereas block-level virtualization operates at the storage physical block layer on LUNs. This type of virtualization can be implemented in operating system software at the logical volume manager layer or in devices inside the storage area network (SAN).

One common form of block-level virtualization is block I/O redirection. Disks are addressed by using unique logical unit number (LUN) identifier and an offset from the start of the LUN; this address is commonly referred to as the logical block address (LBA). The block I/O redirection layer presents virtual LUNs that are mapped to physical LUNs using metadata. This allows the block I/O redirection layer to perform tasks such as physical disk mirroring, striping, and transparent data migration. When an I/O is received by the redirection software device, the LBA is analyzed and the I/O is redirected to physical devices based on metadata in the block I/O redirection layer.

In any enterprise-class storage infrastructure, it is common practice to use some kind of block-level virtualization, at a minimum, in the operating system logical volume manager layer.

Scale out and grid software

Scale out and grid implementations allow multiple physical entities to act as one common storage server with seamless access to data. There are implementations at the block level and at the file level.

At the file level, grid software can offer CIFS and NFS access over an IP network to the same set of file data to one or more clients across a set of multiple server nodes that access a common storage back-end. This is the functionality offered by IBM Generalized Parallel File System (GPFS) and IBM SOFS. This functionality is useful when the ECM system has to serve data at very high speeds to multiple clients, such as in video streaming environments. IBM SOFS also offers integrated data movement between storage tiers, as well as storage mirroring and file snapshot functionality.

Another kind of file-level grid software offers services to replicate files between geographically dispersed nodes and different storage tiers based on user defined file policies. This allows you to place data where it is most likely to be accessed. This function is offered by the IBM Grid Medical Archive Solution (GMAS).

At the block level there are solutions to aggregate multiple small storage nodes into one overall storage device that externalizes LUNs. These kinds of solutions offer constant performance when scaling the number of nodes and additional functionality such as data striping across nodes for performance and data mirroring for availability. One such solution is IBM XIV® grid storage device. This implementation is on the borderline between software and hardware.

These kinds of solutions all allow to scale out the storage infrastructure using off-the-shelf industry standard components.

4.5.3 Storage infrastructure hardware

We now discuss the physical devices that can be used to store data in an ECM environment. As with the storage software, some of these implementation are borderline between hardware and software.

In particular, we discuss disk and tape storage devices, NAS, and protected storage, and mention storage networking.

High performance disk storage

High performance disk storage devices and disk storage systems are commonly used in IT data centers. These devices offer enterprise-class storage characteristics for reliability, availability, and performance, and that is why they usually cost more, on a per GB basis, than low cost storage. Examples of the kind of characteristics that we find are dual redundant controllers, dual paths to data, dual power supplies, high performance disk drives, read cache memory, and write cache memory backed by battery. Enterprise class storage systems usually offer a range of data protection features, from local snapshot copies of data inside the storage system to replication of data to a remote storage subsystem. They often offer synchronous or asynchronous replication to the remote site and at the same time maintain data consistency.

High performance disk storage systems that present LUNs to servers are often, but not necessarily, connected to servers using FC SAN switches and connectivity. They could also be connected using protocols such as iSCSI.

This kind of device is useful for storing data that requires high performance such as the ECM database, which contains metadata that usually requires good random I/O response times.

High performance disk storage can also store the ECM system's object data, if the data requires the performance. If the ECM data does not require the highest levels of performance, or it is not cost-effective to store this data on high performance devices, then consider low cost disk storage in the solution.

Low cost disk storage

Low cost disk storage systems, as the name implies, are cheaper on a per GB basis than high performance disk storage. The lower cost is usually obtained by using less expensive disk devices, which often use SATA technology disks that are less expensive per GB than those used in high performance storage systems. The lower cost comes at the expense of performance because lower cost storage devices usually have lower performance characteristics. SATA disk storage is quite adequate for storing infrequently accessed files and for files that are accessed sequentially with low contention, which are typical access patterns for ECM data. SATA disk devices are usually not recommended for storing the ECM database because it has a random IO access pattern. The connectivity of these devices, as for the high end devices, is usually through the FC protocol; iSCSI is also applicable because often the performance requirements are not as stringent as for high performance disk.

Most high end storage devices offer the possibility to intermix different disk storage technologies inside the same box, so data with different access requirements can be placed on different storage tiers inside the same physical disk system. It can be advantageous to intermix different disk storage tiers in the same system because it gives you the possibility to use storage replication functionality at the overall system level. Therefore, if you license remote replication features, you can use these features for any storage tier in the storage system. The tiered storage disk system can contain both the ECM database metadata on high performance storage and ECM data on low cost storage, and both can be replicated together with consistency.

Network-attached storage

The term network attached storage (NAS) refers to a storage device that is accessed through a LAN network connection using a transport protocol such as TCP/IP. Traditionally, NAS devices typically serve as dedicated file servers for heterogeneous clients connected to the LAN network, but recently the boundaries between SAN and NAS storage have started to blur and there are products in the market that offer connectivity using both file-level and block-level protocols.

NAS provides an interesting option when you do not require the most stringent performance and would like to avoid the cost of setting up a dedicated SAN.

Many NAS devices offer additional function to perform local snapshot copies of data or remote mirroring to a separate NAS device. The snapshot functionality is often space-efficient because NAS devices use space only for new updates to the data. This can offer an interesting solution for scheduled backup operations because it requires minimal or no ECM application downtime.

Removable tape storage

Removable tape storage is a proven and consolidated technology for low cost data storage with good performance characteristics. An automated tape storage system usually consists of one or more tape drives to read and write the data, tape cartridges that store the data, and a tape library for cartridge storage capacity and a robotic mechanism to mount tape cartridges into the drives. In general, tape has the following performance characteristics:

- **Access to first byte of data:** Relatively slow because of tape mount and positioning times
- **Sequential data reads:** Good because high end tape devices offer sustained data rates in the order of 100 to 200 MB/sec

Tape is often more energy efficient than disk because it only uses energy when reading or writing the data; when not in use, the tape drives do not consume any energy. This is of particular interest in ECM solutions where large amounts of data might have to be stored for long periods of time for audit and regulatory reasons, but access to the data is very infrequent. It is not recommended for data that is frequently accessed because of the delays in accessing data, due to mount and positioning times.

Tape is also a cost effective media for data protection both for local backup operations as for remote vaulting for disaster recovery. For disaster recovery, tapes can be produced at the primary data center and shipped periodically to a remote site or vault and, depending on the volume of data, this could be more cost effective than sending it offsite electronically.

Protected storage devices: WORM

Protected or WORM storage devices refer to physical storage devices that do not allow you modify or delete data once it has been stored. These kinds of devices are required to comply with certain regulations, and they also offer protection against accidental or intentional tampering with the data. There are multiple terms that are commonly used to define these devices, such as:

- WORM (write once read many) storage devices
- WORN (write once read never) storage devices
- Non-erasable non-rewritable storage
- Protected storage devices

There are two broad classes of WORM storage media and devices. Physical WORM devices such as optical DVD-R or CD-R rely on the physical properties of the materials with which they are manufactured to stop you from overwriting or altering the data. Logical WORM devices rely on software or microcode (software implemented in hardware) to provide the WORM protection.

Optical WORM media is not commonly used because of its low capacity and performance characteristics. Today, most WORM implementations are based on disk or tape media. The tape media implementations rely on microcode in the drive and in a chip on the tape cartridge to ensure that the cartridge cannot be overwritten. Disk media implementations use either programmatic APIs as in the case of the IBM DR550, or a network file system interface such as CIFS or NFS, as used by the IBM N series and optionally by the IBM DR550.

Good disk or tape logical WORM implementations offer multi-tier storage for data and the possibility of storing multiple copies of data for data protection and disaster recovery purposes.

Storage network interconnects

A storage area network (SAN) is the layer that is used to connect servers to storage devices. Devices attached to a SAN appear to the server as if they were locally attached. Devices on a SAN are commonly addressed using the SCSI command set protocol and the FC transport protocol. Other transport protocols can be used such as iSCSI (SCSI transported over TCP/IP) or SCSI over Infiniband, but the FC transport protocol is the most common.

FC SANs are usually implemented using FC switches and directors that allow multiple servers to access and share multiple storage devices. From a performance perspective, an FC SAN is the optimal connectivity layer between servers and underlying storage devices. The use of an FC SAN is also dictated by the kind of storage devices that are chosen and what interfaces they offer.

4.5.4 Functional relationships between components

When designing a storage solution for an ECM environment, it is important to understand the functional relationships between the ECM layer and the storage layer. Figure 4-3 shows an overview of how an object is stored and indexed in a generic ECM system.

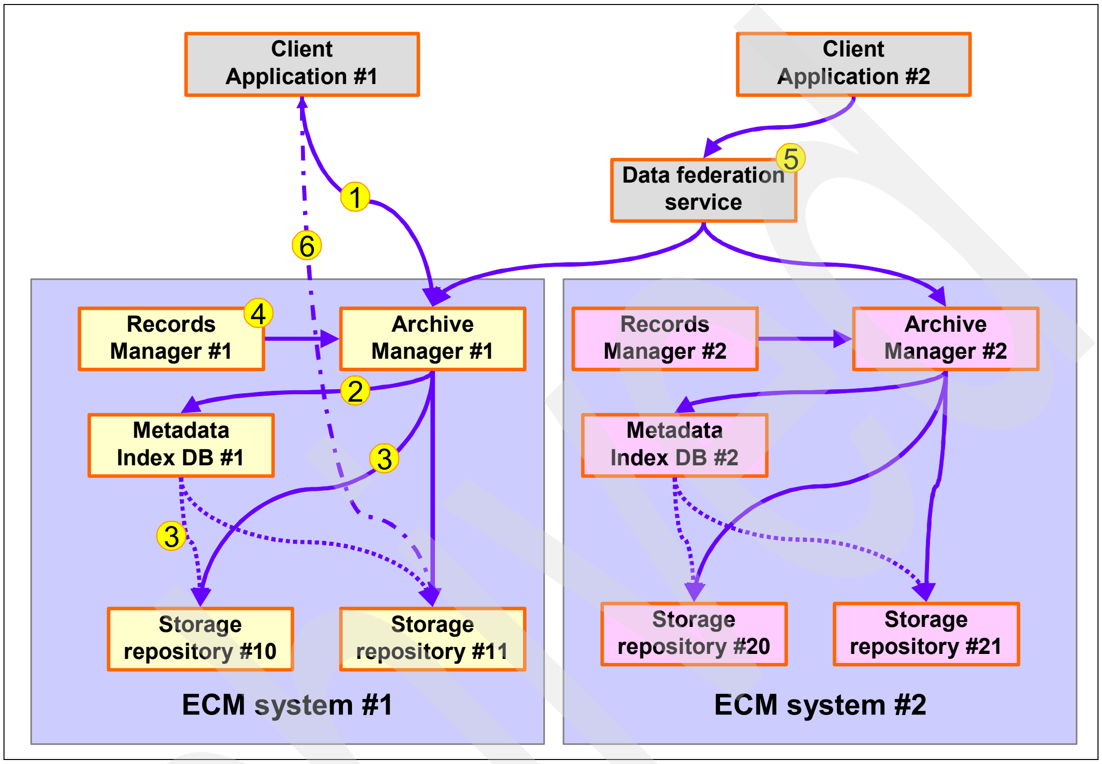


Figure 4-3 Generic ECM system to storage model

A client application accesses the ECM system#1 to store or retrieve objects, the client application will contact an archive manager #1 service (1). The archive manager #1 service is responsible for indexing the data objects (2) in a metadata index database #1. For each object, the metadata index database contains the storage repository locations where the data is stored (3) allowing the archive manager service to store new data or return stored data to the application (3). Each archive manager has one metadata index database and can have one or more storage repositories connected to it. The archive manager #1 can apply retention policies and rules to the data under the direction of the records management service #1 (4).

Another client application, application #2, might have a requirement to access data from multiple archive managers in two different ECM systems, ECM system #1 and system #2. In this case it would use a data federation service (5) that takes control of the request and forwards it to multiple archive managers, based on data mappings defined at the federation service layer. The data federation service allows you to also access data stored in multiple different ECM systems.

In some cases it can be useful to access the data directly from the storage repository (6). In this scenario the client application can locate the information using the archive manager and the archive manager will return a pointer to the data to the requesting client. This pointer could be an http address or a file name. The client can then access the data directly from the storage repository bypassing the archive manager. This is useful when handling large volumes of data such as video and other streaming media.

Data access characteristics can change over time so there might be a requirement to change the storage media tier for an individual data object or a group of objects at a predefined moment in time or when a specific event occurs. This action can be performed in multiple ways and at multiple levels, as shown in Figure 4-4.

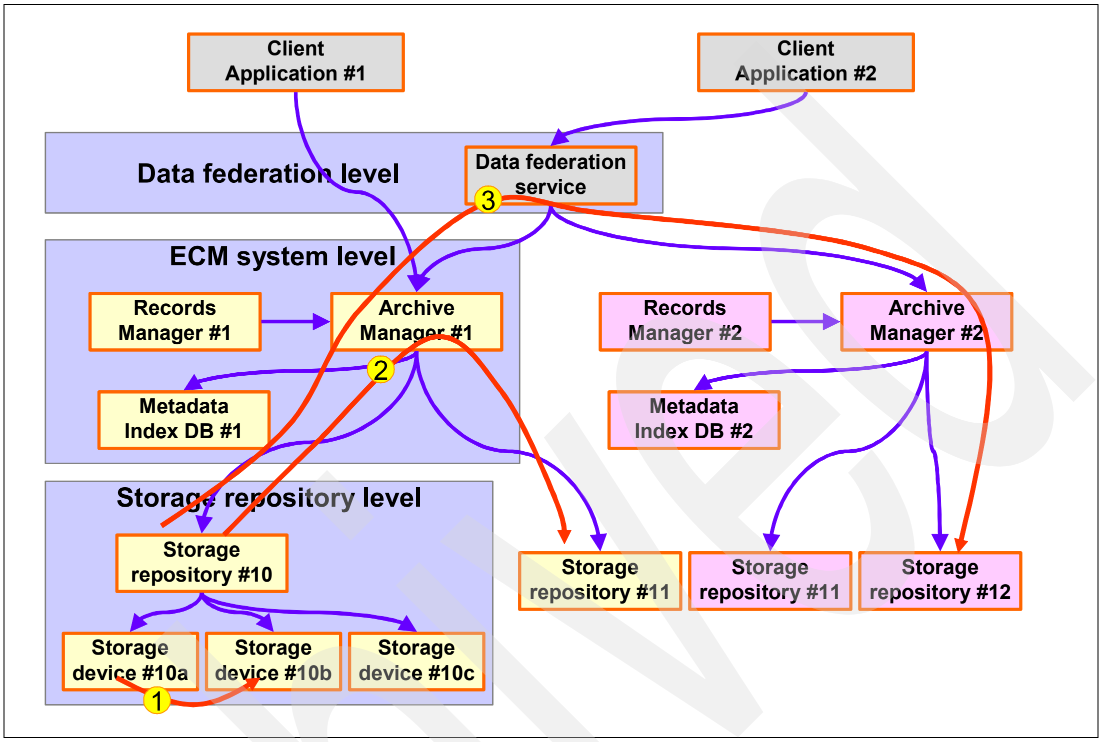


Figure 4-4 Data object movement possibilities

Data can be moved between tiers at the storage repository level as illustrated in (1). In this case the data is moved between storage device #10a to device #10b by some entity, usually software based. The ECM application at the upper level is not aware that data movement has happened, as the action is performed transparently to the ECM layer. An example is policy based HSM software acting at the file system level - the HSM software will move the data to a new storage device transparently and will handle data retrieval transparently when the application requests the data. The ECM system generally does not have direct control of which data is moved nor when it is moved. Some storage implementations allow the applications, such as the ECM system, to signal which files have to be moved to a different tier inside the same storage repository, but this function is not commonly used. To perform such an action, you have to customize the ECM system to communicate to the underlying storage repository which objects have to be moved and when.

If the ECM application wants to retain control over what data is moved to a different storage tier and when this data is moved, then it has to directly take control of data movement as shown in (2) where the archive manager service moves data from storage repository #10 to storage repository #11. The ECM application can also replicate the same data between multiple storage repositories so that the data is protected in the case where you lose one storage repository. This is useful for disaster recovery purposes.

There is also a case where data has to be moved between different ECM systems, either different instances of the same ECM software or completely different ECM software implementations. This movement will be done at the data federation level (3). This kind of data movement is not commonly used and is usually reserved for cases where you have to migrate data from one ECM system to another.

4.5.5 Design guidelines and trade-offs

When designing storage components for an ECM solution, there are some design rules to keep in mind for database storage and for object storage. One basic and crucial element in designing storage for the ECM solution is to understand the application's storage requirements in terms of data volume, number of objects, access frequency, retention, and protection requirements.

The business line that owns the application must give an estimate of the data volumes, which is the number of objects that have a requirement to be stored and managed by the ECM system. This number of objects could depend on the number of customer transactions the system must process, for example, the number of payment statements that are issued or the number of insurance claims in a specific period. Application owners must give an estimate of the average object size and number of objects per transaction. The number of objects per transaction depends on the type of application; for example, an insurance claims application might have multiple images and documents associated with an individual claim. Individual objects might have different storage requirements; for example, it might be advantageous to separate large image files and small documents between different storage devices and tiers.

The access frequency to an object must be determined. The object is initially received and stored onto a storage repository by the archive manager service. What is the probability of the object being referenced in the future? This is application dependent; for example, in the insurance claims business, an individual claim can be frequently accessed until it is paid off, then the chance of accessing that particular claim will probably drop considerably, as the object becomes inactive. Consider whether it is advantageous to move inactive objects to lower cost storage, either at the ECM system level or at the storage repository level.

The object's retention criteria will have to be evaluated. Retention depends on business and legal requirements for the specific ECM application; retention is not defined by IT operations. Objects might have to be retained for years or decades, and this is an important factor to consider. If the objects have a low access frequency, and access time requirements for aged objects are not time-critical, then you have to consider tape media in the equation because tape media is often more cost-effective than disk. Lower cost disk is also an option together with tape storage. Plan to have an automated way to move data to the next storage tier. For objects that require long term storage that outlives the storage media itself, also consider and design for migration scenarios: how to move the data to new storage devices with minimal or no impact on the application.

Protection requirements also depend on business and legal requirements for the application. By protection we mean the use of protected, or WORM, storage devices that do not allow stored data to be modified or deleted. Evaluate your business and legal requirements for protected storage, which can be implemented using disk devices, tape devices, or both kinds of devices connected together in storage tiers under the control of the storage repository service.

Database storage

The archive manager metadata index database, also referred to as the *ECM database*, is usually based on an industry standard relational database such as DB2, Oracle®, or Microsoft SQL server. Depending on the implementation, there can be one or more database instances in a given ECM system. The ECM database performance and availability requirements can vary, but in general, the database will require good response times and adequate availability and failover mechanisms. Database performance depends on many factors, important factors are size, database layout on disk, and underlying disk performance. Databases reside on standard disk storage devices, not on tape or protected storage devices.

The ECM database size depends on the number of objects that the database will contain, and the number of objects is driven by business application and retention requirements. The database size for a specific ECM software implementation can be estimated using that ECM product's sizing tools and guidelines. These are very product dependent and might vary between product releases.

When in use, ECM databases are usually accessed randomly so you will have random I/O going to the underlying disk device, random IO performance is helped by high end disk devices and SATA storage devices are not recommended for disk storage. Attempt to spread the database I/Os on as many physical disk spindles as possible and also separate the database physical storage from database log physical storage.

The ECM database might also require high availability and this dictates a requirement for enterprise class storage devices. Storage systems with RAID protection are a de-facto standard and you might want to consider storage that offers remote replication functionality.

Object storage repository

The type of storage to use for archive manager object storage repository is highly dependent on application requirements and the underlying data access patterns. Disk storage devices must be considered as a first tier repository for the data, in this way data can be stored by the archive service without the delays that would be incurred if you had to mount a tape cartridge.

The object storage repository typically stores unstructured data, such as documents and images, in the form of files that are accessed as a whole, that is read from start to end. This is different from databases where you open the database files and then randomly access individual data blocks. Object storage repositories are usually accessed in a *skip-sequential* manner. The ECM application determines the specific file to access, and this can be any file that is stored so we have a random access to the start of the file and then the file is read sequentially from start to end and returned to the application. Larger average object sizes will result in a greater sequential character of the workload.

Another important factor is the access density to the object storage repository. Access density is often defined as *the number of I/Os per second per gigabyte of storage*. Access density depends on the way the application references the data, for example, new files might be frequently accessed in the first year and then very infrequently in the remaining nine years they are kept for. If the disk storage contains only new files it will have a higher access density than if it contained an even mix of new files and files over one year in age.

Mid-range disk storage devices and subsystems with SATA disks often offer a good price performance trade-off for storing ECM data. They certainly must be selected if you choose to have a disk-only object storage repository. On the other hand if you keep only highly referenced recent data on disk and migrate the older data to a media such as tape, then you might have to consider higher performance disk storage devices.

4.5.6 High availability

High availability of the underlying storage environment must be considered when high availability of the ECM solution is required. High availability is often implemented through redundancy of individual hardware and software components and transparently moving load from a failed component to the remaining components. In storage terms this can be implemented with features and configurations such multiple network paths to the storage devices, dual controllers in disk storage subsystems, RAID protection inside the disk storage system and multiple tape drives inside a tape library.

High availability at the disk storage system layer can also involve multiple systems with hardware or software replication functionality to keep data aligned between the systems. Care must be given to keeping database and object storage in sync, so as to avoid cases where the database references an object that is not present on object storage. This is particularly important in cases such as backup and recovery where data base and object storage could be recovered to different points in time. In this case we would have data inconsistency between database and objects and the inconsistency would have to be resolved by hand.

The database and object storage volumes must be grouped, protected, and recovered as a whole. This is called data consistency, which can be implemented by quiescing the ECM application and its database and taking a backup. In high end storage systems that offer replication functionality to a remote site, data consistency is implemented through the use of consistency groups, which guarantee that data at the remote site is consistent at all times, even in case of individual failures in the network containing the replicated storage devices.

Server clustering is an important component in high availability. If a server fails, the ECM archive service can be restarted on a different node by clustering software. The ECM application requires access to the underlying storage repositories and storage devices, so the repositories and devices must be accessed by the remaining cluster node. In general, with disk storage locally connected to the server (DAS), the cluster software will have to mount the volume on the failover server, whereas in the case of network attached storage (NAS), the same shared resource can be visible concurrently from multiple nodes. This facilitates recovery because you only have to restart the application on the failover node, providing that the volumes have previously been mounted.

Tape storage can be integrated into a high availability solution by using an intermediate software layer such as Tivoli Storage Manager, which can access the underlying tape devices. Access to the tape device would be through this software layer so the application would see the tape storage as a service that is brokered by Tivoli Storage Manager.

4.5.7 Disaster recovery

Disaster recovery is an extension to high availability that is used to resume operations in a remote site when the whole primary site is unavailable or destroyed. To perform recovery from a disaster, we require three main elements:

- **Site:** A place where we can perform recovery
- **Data:** A usable copy of the original data
- **IT infrastructure:** Servers, storage, and network elements on which to recover our data and restart our applications

From a storage perspective, we have to consider data. A *usable copy of original data* means that the data must be consistent: The copy of the database must be in sync with the copy of the object data.

The first thing to choose is the method of producing data for disaster recovery. This is dictated by business requirements and will dictate the RPO (amount of data loss) and RTO (time to recover) times. Low RPOs, under a day, require some kind of storage replication mechanism to send the data to the remote site. RPOs greater than or equal to one day can be satisfied from offsite tape rotation mechanisms. RTOs from tape are in the order of days to weeks depending on the amount of data to be recovered, so short RTOs of a day or less will require that data on disk at the primary site be available on disk in the secondary site.

If you use a storage management product such as Tivoli Storage Manager, it is possible to define configurations where DR data is all stored on tape, and in a DR scenario, only the metadata index database is restored, whereas the object data is accessed directly from tape. This type of configuration offers lower performance in the disaster recovery site and could be acceptable when the number of object retrievals is projected to be low.

If requirements dictate it, the disaster recovery copy of data can be stored on protected storage devices so that if a disaster hits the primary copy of data, the remaining copy is still protected. You could also decide that the primary copy of data is protected and the secondary one is not, depending on business requirements.

Sending sensitive data to a remote site can require the use of encryption techniques to avoid the risk of security exposures if the information falls in the wrong hands. Encryption must be implemented for data that has to go outside the IT datacenter. If data is sent off-site on tape, then you have to evaluate tape encryption products. If data is sent to the remote site over the network, then consider using network encryption.

It is also possible, but not common, to have data replicated at the archive manager layer in the ECM application. Some ECM implementations offer this kind of functionality and allow data to be replicated by the archive manager service between two separate storage repositories; one could be local and on remote. The archive manager index database will have to be replicated using other techniques or backed up periodically to the remote site.

4.5.8 Storage data migration

Storage data migration scenarios must be considered when designing the ECM storage infrastructure, especially in cases where data retention periods are greater than the expected lifetime of the underlying storage devices. Plan flexibility into the overall solution so that the introduction of new storage hardware components is not too disruptive to the application. Also, consider block level storage virtualization as an enabling technology to assist with the migration to new storage devices, because it can enable transparent data migration between different storage devices. Storage management products such as Tivoli Storage Manager and the DR550 can insulate the user from the underlying storage and facilitate data migration and data replication in multiple storage devices and across storage tiers. NAS storage can also be virtualized and transparently migrated to new storage devices using file-level virtualization solutions.

If you do not regularly use a disk virtualization layer, you can always use data migration software products on a temporary basis when it becomes necessary to migrate data.

Storage data migration and virtualization techniques allow you to migrate data between storage devices without impacting the ECM system layer. The ECM system is not aware that the data has been migrated to a new device.

There might be cases when data has to be migrated across different ECM systems or between ECM archive managers.

Most enterprise ECM systems allow you to set up data migration procedures to migrate data between different ECM systems. This kind of functionality is based on data federation functions that allow you to map data to multiple ECM systems. Data migration using data federation requires application level changes and actions to the ECM system. It can be achieved, but is outside of the scope of the Storage Reference Architecture.

5

Storage products overview and matrix

In this chapter, we describe the various IBM storage products and solutions that can be used in an ECM environment. We relate storage products to information infrastructure themes such as availability, retention, compliance, and security.

We discuss the following topics:

- IBM disk systems
- IBM tape libraries and drives
- Network Attached Storage (NAS)
- SAN Volume Controller
- Storage data management products, such as Tivoli Storage Manager and DR550
- File systems, such as IBM Generalized Parallel File System (GPFS)
- Storage management products, such as IBM TotalStorage Productivity Center

5.1 Information infrastructure components

In this chapter we introduce and discuss the various storage products that IBM offers and show how each one addresses the various information infrastructure requirements. The information infrastructure requirements derive from mapping business requirements to functional areas and capabilities presented in each single product and product line.

The four functional areas or themes that make up the information infrastructure are:

- **Information Availability:** Related to high or continuous availability, and data access performance
- **Information Security:** Related to data protection, encryption, sharing, and avoidance of unauthorized access
- **Information Retention:** Related to placement and movement of data between the most cost-effective storage tiers, and timely deletion of old data
- **Information Compliance:** Related to risk and data integrity, protected, or unmodifiable storage

The matrix in Figure 5-1 maps these four functional areas or themes and how these themes are addressed by individual storage products or components. Each theme is in turn broken down into facets that reflect storage product, or product component, functions, or capabilities relevant to that specific theme. The storage products are grouped into product families based on the type of storage product.

	Information themes	Information Availability					Information Retention					Information Compliance		Information Security
Storage product family	Product features - Theme facets	High performance	Virtualized storage	Simplified management	Replication and recovery	Automated storage	Tiered storage	Policy based archiving	Offline storage	Low cost storage	Deduplication	Governance and policy automation	Protected storage	Encryption
Disk	DS3000									x				
	DS4000	x			x		x			x				
	DS8000	x			x		x							
	XIV	x		x	x					x				
Tape	TS drives								x	x			x	x
	TS libraries								x	x				
	TS VTLs				x				x		p			
NAS	N Series		x	x	x	x					x		x	
	SOFS		x		x	x	x	x	x	x		x		
	GAM		x		x	x	x	x	x			x	x	
Block virt.	SVC	x	x		x	x	x							
Storage data management	TSM		x		x	x	x	x	x	x	p	x		x
	SSAM		x		x	x	x	x	x	x		x	x	x
	DR550		x		x	x	x	x	x	x		x	x	x
File systems	GPFS		x		x	x	x	x	x	x		x		
Storage management	Productivity Center for Data			x		x	x		x					
	Productivity Center for Fabric			x		x								
	Productivity Center for Disk			x		x			x					
	Productivity Center for Replication			x	x	x								
	SSPC			x										

Figure 5-1 Storage theme to storage product matrix

The matrix shown above is not necessarily complete; there might be other IBM products that offer specific function for a certain information theme and facet. It is used as a broad positioning outline to help you understand which products are available and what functions and benefits are offered by each individual product.

In this broad classifications we have included product families such as Tivoli Storage Manager and N series, which offer a range of diverse functions that can cover many functional and non-functional requirements.

5.2 Products for the information infrastructure

This section is not intended to provide a comprehensive catalog of all available IBM storage products and solutions, as this information is readily available in other publications. Its intent is to reflect the product components we commonly use for ECM storage solutions. We discuss both hardware and software products.

5.2.1 Disk storage systems

IBM offers a comprehensive family of disk storage products that is adaptable to most requirements. The IBM System Storage disk products family, also referred to as the DS family, covers a wide spectrum of possible implementations, from entry-level storage implementations to large enterprise deployments that can span multiple sites.

The IBM System Storage DS family includes products to cover a range of requirements:

- The IBM System Storage DS6000 and DS8000 series enterprise class high performance servers, for the most demanding workloads and greatest scalability and reliability
- The IBM System Storage DS4000 series of midrange systems
- The IBM System Storage DS3000 series of low price entry systems

These families are further complemented by a range of expansion enclosures to expand the disk storage capacities of individual systems to hundreds of terabytes.

The IBM System Storage product line also offers a comprehensive range of capabilities such as advanced copy services, management tools, and virtualization services to manage and protect data.

Figure 5-2 shows the range of DS disk storage systems, they are discussed in more detail in the following sections.

Figure 5-2 The IBM System Storage DS family

DS3000 series

The DS3000 are external storage systems equipped with SAS drive technology and the flexibility of iSCSI, SAS, or FC server connectivity small to medium environments.

IBM System Storage DS3000 series is designed to provide the flexibility of the Serial Attached SCSI (SAS) technology. There are multiple models based on the type of connectivity that they provide.

The DS3200 is a system for entry-level shared common direct attach storage supporting up to a maximum of three servers. It offers one to three 3-Gbps SAS host ports per controller and is an entry point for 2-node clusters.

The DS3300 is well suited for low-cost storage networking environments, it offers two 1-Gbps iSCSI host ports per controller to build an IP based SAN solution.

The DS3400 is an ideal system for SAN implementations in small environments, it offers two auto-negotiating 4-Gbps host ports per controller and can be used as a direct-attach or FC SAN solution. It is suitable for building new SANs or low-cost additions to an existing SAN.

All the DS3000 series models supports the EXP3000, a Serial Attached SCSI (SAS) based expansion enclosure that can increase capacity up to 48 disks without the necessity for additional investment in controllers. Common features across all the DS3000 family are:

- 2U enclosure supporting up to 12 SAS and or SATA drives
- Dual-active RAID controllers with mirrored, battery-backed cache with 512 MB per controller standard and 1 GB per controller optional
- Redundant, hot-swappable controllers, drives, power supplies, and fans
- Intuitive DS3000 Storage Manager software
- Support for up to 16 Storage Partitions
- Support for FlashCopy® and Volume Copy

The DS3000 series is a good entry solution for departmental and distributed environments. It also offers basic data protection services such as flash copy and volume copy.

For more information about the DS3000 disk series, refer to the following Web site:

http://www-03.ibm.com/systems/storage/disk/entry/index.html

DS4000 series

IBM System Storage DS4000 series can help handle high-performance or high-capacity storage requirements. The IBM System Storage DS4000 series uses common storage management software and high-performance hardware design, helping to provide customers with enterprise-like capabilities at a low cost. High-availability, multi-platform support, a broad range of open OS support and comprehensive management tools all help you adjust to changing storage requirements and challenges.

IBM System Storage DS4000 series are disk storage products using redundant array of independent disks (RAID) that contain the Fibre Channel (FC) interface to connect both the host systems and the disk drive enclosures. The DS4000 series of disk storage systems are an IBM solution for mid-range and departmental storage requirements.

Figure 5-3 shows the various systems available in the IBM System Storage DS4000 series family.

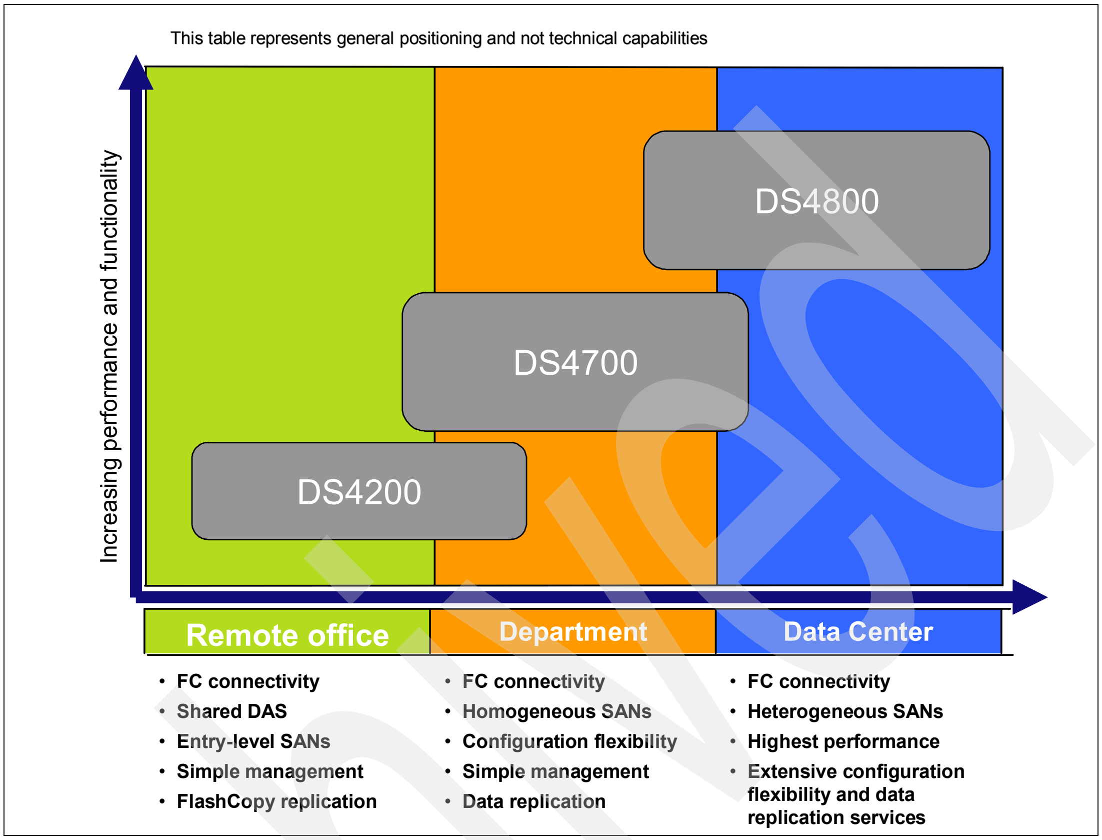


Figure 5-3 The DS4000 product family

All models in the DS400 family offer 4 Gbps FC connectivity. The DS4800 and DS4700 offer both FC and SATA drive intermix support, whereas the DS4200 offers only SATA drive support. All models offer switched-loop back-end for optimal performance. These are common functions offered by all DS4000 systems:

- 4 Gbps FC host interfaces that auto-negotiate 1, 2, or 4 Gbps
- Dual-active RAID controllers with mirrored, battery-backed cache
- Redundant, hot-swappable controllers, drives, power, ESMs, and fans
- Unlimited global hot spare disks with automated failover and failback
- DACstore protects configuration metadata on each drive
- Hardware XOR engines for efficient RAID parity calculations
- Switched-loop architecture for enhanced diagnostics, lower latencies, and linear performance scaling
- Feature-rich management software with online administration and unparalleled configuration flexibility

The DS4000 family offers a range of functions for protecting data and creating split mirror copies. These are:

- FlashCopy for local space efficient copies
- VolumeCopy for local copies of data
- Enhanced Remote Mirroring for copying data volumes between two different DS4000

FlashCopy

The DS4000 FlashCopy allows you to create a point-in-time (PiT) image of a logical drive; it is the logical equivalent of a physical copy. The copy is performed instantaneously and requires less disk space than a full copy. The resulting copy is mappable to any host and it can be read from, or written to. It is used primarily for PiT backup image, as a source for file and volume restoration and for data mining and analysis. The DS4800 supports up to 1,024 maximum FlashCopy images whereas the DS4700 and DS4200 support up to 512.

VolumeCopy

The DS4000 VolumeCopy allows you to create complete point-in-time replication of one logical drive (FlashCopy image) to another within a disk system; the target drive is also referred to as a clone. It is used for full PiT data set available for analysis, mining, or testing as it does not degrade performance of production logical drive. It can be used to redistribute data for performance and capacity optimization, as a tool for technology migration to new drives or to move logical drives to more desirable RAID arrays. This feature is offered on all DS4000 systems.

Enhanced Remote Mirroring

The DS4000 Enhanced Remote Mirroring offers storage subsystem based data replication. It is designed to provide ongoing, real-time replication of a logical drive from one DS4000 storage subsystem to another. Three mirroring modes are available:

- Metro Mirror for a synchronous copy of data
- Global Copy for an asynchronous copy of data, used in data migrations
- Global Mirror for an asynchronous and consistent copy of data

Its primary uses are disaster recovery, centralized data backup, and centralized data mining and analysis. The DS4800 supports 128 maximum mirrors whereas the DS4700 and DS4200 support at most 64 maximum mirrors.

DS4800 series features

The DS4800 is the flagship system designed for enterprises with compute-intensive applications and large consolidations. It has the highest performance of all the DS4000 family. The DS4800 offers premium features such as up to 512 partitions, FlashCopy, VolumeCopy and Enhanced Remote Mirroring. The DS4800 can scale up to 224 TB with both FC and SATA disk.

DS4700 series features

The DS4700 is a fully-featured systems designed for midrange environments requiring high-end functionality and performance value. It offers 4 Gbps FC interfaces and the ability to intermix FC and SATA drives. The DS4700 offers premium features such as up to 128 partitions, FlashCopy, VolumeCopy and Enhanced Remote Mirroring. The DS4700 can scale up to 112 TB with either FC or SATA disk.

DS4200 series features

The DS4200 is a fully-featured disk system designed for entry-level and secondary storage requirements. It offers 4 Gbps FC interfaces and can only host SATA II drives. It offers high end functionality at an attractive cost per gigabyte.

The premium features offered by the DS4200 include up to 128 partitions, FlashCopy, VolumeCopy and Enhanced Remote Mirroring. The DS4200 can scale up to 112 TB with SATA disk.

DS4000 Storage Manager

All DS4000 models are managed by a common storage management software called the DS400 Storage Manager, that offers many features and functions for DS4000 systems. It allows you to lower storage management costs because it offers:

- Centralized administration of all DS4000 systems
- Common interface across all DS4000 products and platforms
- Robust functionality with intuitive GUI and wizards
- Fully-integrated advanced functionality (premium features)
- Configuration flexibility helps creates outstanding storage utilization

DS4000 systems, together with the DS4000 storage manager, offer functions and features to help ensure that data is always available. These are:

- EXP HotAdd Technology for online capacity expansion, configuration and LUN access
- Dynamic Array Expansion (DAE) to add drives to existing array without stopping application (restripes logical drives, defrags array)
- Dynamic Volume Expansion (DVE) to add available capacity to existing logical drives (this feature is OS dependent)
- Dynamic RAID Level Migration (DRM) to change the RAID configuration of an array (without relocating data)
- Dynamic Segment Size Migration (DSS) to change the data stripe size (without relocating data)
- Dynamic mode switching to change between remote mirroring modes

From the information themes view, the DS4000 family offers both high performance and low cost characteristics based on the type of disk drive and subsystem model you choose. It offers replication and recovery functions through the Enhanced Remote Mirroring features and allows for tiered storage by intermixing FC and SATA drives in the same subsystem.

For more information about the DS4000 disk series, refer to the following Web site:

http://www-03.ibm.com/systems/storage/disk/midrange/index.html

DS6000 and DS8000 series

The family of IBM Enterprise Disk Systems offers a range of scalable solutions to address various enterprise storage requirements. Leveraging IBM leading technology, the Enterprise Disk Systems provide a significant choice in functionality, performance, and resiliency.

We will illustrate features and major characteristics of the following enterprise disk storage products:

- IBM System Storage DS6000 Series
- IBM System Storage DS8000 Series

The IBM Enterprise Disk Systems deliver high-performance, high-availability storage, with flexible configuration for different business requirements. The IBM System Storage DS6000 and IBM System Storage DS8000 deliver an enterprise storage continuum of systems with shared replication services and common management interfaces. The DS6000 and DS8000 series systems are designed to help simplify the storage infrastructure, support business continuity, and optimize information lifecycle management.

The DS6000 offers enterprise-class functionality with modular design and an attractive entry-level pricing for medium and large businesses. The DS6000 series helps simplify data management and enables easy scalability, which allows accommodation of the continuing exponential data growth; it brings proven enterprise class technology to a modular package.

The IBM System Storage DS8000 series is designed for the most demanding, mission critical environments requiring the highest level of availability. The DS8000 series is designed to set an entirely new industry standard for high-performance, high-capacity by delivering a dramatic leap in performance and scalability. The DS8000 family has two model families:

- The 2-Way DS8100 Turbo models offer:
  - Two dual POWER5+™TM based processor servers
  - Up to 128 GB Cache
  - 8 to 64 4Gb or 2Gb FC/FICON® – 4 to 32 ESCON® Ports
  - 16 to 384 HDD
  - Intermixable disk drive packs
  - FC and FATA drive support
  - Physical capacity from 1.1 TB up to 192 TB
- The 4-Way DS8300 Turbo models offer:
  - Two four processor POWER5+TM based servers
  - Up to 256 GB Cache
  - 8 to 128 4Gb or 2Gb FC/FICON – 4 to 64 ESCON Ports
  - 16 to 1024 HDDs
  - FC and FATA drive support
  - Physical capacity from 1.1 TB up to 512 TB

The DS8000 comes in a frame enclosure, shown in Figure 5-4, that can be expanded to five frames for greatest scalability.

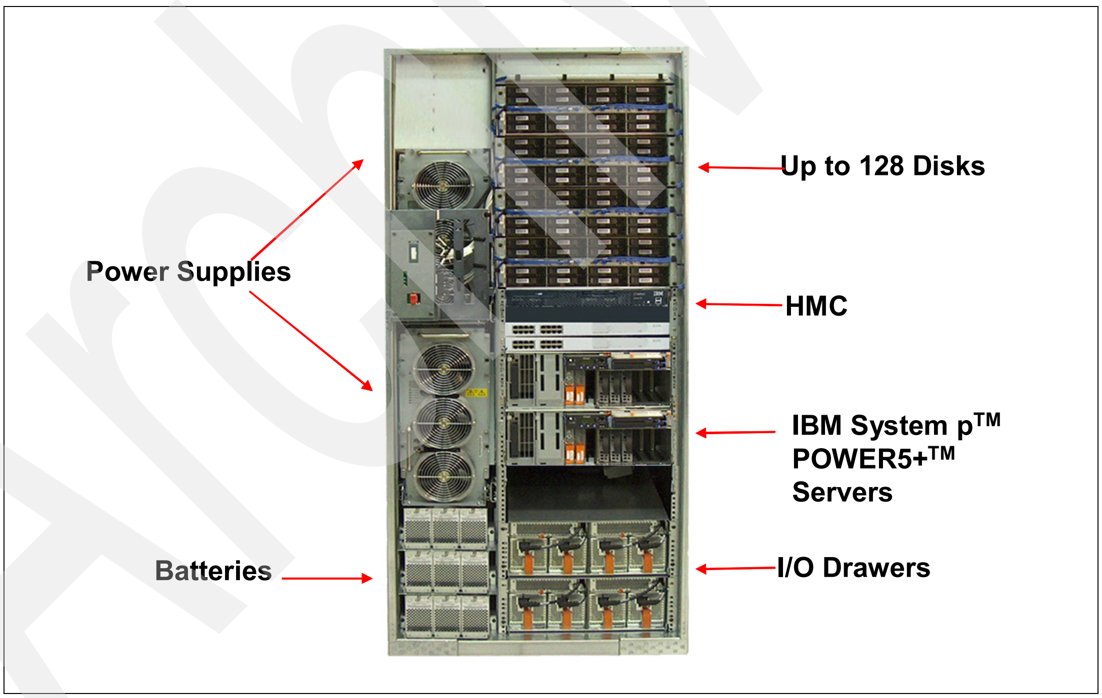


Figure 5-4 Internals of a DS8100

The IBM System Storage DS8000 series of high-performance, high-capacity storage systems is designed for scalability, resiliency and overall total value. The DS8000 Turbo models are the newest members of the DS8000 series. They provide even greater performance and scalability utilizing IBM POWER5+ processors and 4 Gbps FC/IBM FICON connectivity.

The DS8000 offers front-end, single port performance of up to 400 MBps, and very high back-end performance. The DS8000 series is designed to deliver greater total throughput for typical production workloads than other enterprise disk storage platforms. The DS8000 series also includes tiered-storage capabilities that allow for considerable flexibility in cost and performance disk drive options. Optimizing storage this way can help control the high costs associated with the exponential growth of data.

The DS8000 series also provides excellent features to simplify storage infrastructures by allowing two independent, virtual storage system logical partitions (LPARs) to run on a single system. With leading edge performance, virtualization and intelligent automation, the DS8000 series leverages decades of IBM innovation in high performance, highly available computing systems.

The DS8000 support continuous operations for mission-critical workloads. It helps reduce operational complexity through consolidation with unique storage system logical partitions (LPARs). Enterprise-class resiliency features helps avoid single points of failure and provides high availability with 2-site and 3-site remote mirroring for support of 24x7 environments.

It offers lower energy consumption per unit of space through a tiered storage environment and choice of Fibre Channel and Fibre Channel ATA disk drives. Models are field upgradeable and dual 2- and 4-way models offer excellent investment protection.

The DS8000 is designed for outstanding performance, the DS8000 series has the power necessary to support up to 512 TB of physical storage capacity to help accelerate data input/output. Modular hardware design also helps deliver balanced performance throughout the storage system to help optimize overall throughput.

Copy services

The DS8000 series is designed for resiliency and business continuity. IBM FlashCopy point-in-time copy functions can support online backups for high application availability and continuity of operations, while IBM Metro Mirror, Global Mirror and Metro/Global Mirror business continuity solutions are designed to provide the advanced functionality and flexibility to tailor a business continuity environment for almost any recovery point or recovery time objective.

FlashCopy is for near-instantaneous logical copies that minimizes copy establish times. Standard features are:

- NOCOPY function for copy on first write optimization
- Incremental FlashCopy support
- z/OS Data Set FlashCopy
- Multiple Relationship FlashCopy
- FlashCopy Consistency Groups
- FlashCopy anywhere within same DS8000 with no disk position restrictions

Remote Mirroring for external replication between systems. The available types of remote mirroring are:

- Metro Mirror for synchronous copies, remote site data currency, metro distance, consistency groups
- Global Mirror for asynchronous protocol, remote site near-data currency, unlimited distance, consistency groups
- Global Copy for asynchronous protocol, period point-in-time currency, unlimited distance
- Metro/Global Mirror 3-site with A-to-B synchronous + B-to-C asynchronous
- z/OS Global Mirror optimized for z/OS, asynchronous protocol, remote site near-data currency, unlimited distance - XRC

- z/OS Metro/Global Mirror three site solution using Metro Mirror + zSeries® Global Mirror

Physical capacity

With the DS8000 Turbo Models, physical capacity can be configured as RAID-5, RAID-10, or a combination of both. RAID-5 can offer excellent performance for most applications, while RAID-10 can offer better performance for selected applications, in particular, high random write content applications in an open systems environment. RAID-10 combines RAID-1 (mirroring) with RAID-0 (striping). Each drive within a RAID-10 array is mirrored to a second drive within the array, and to optimize performance, data is striped across the drives within the array.

Storage Manager

The IBM System Storage DS Storage Manager is a single integrated Web-based graphical user interface (GUI) that offers:

- Offline configuration: Allows a user to create and save logical configurations and apply them to an online DS8000 series system.
- Online configuration: Provides real-time configuration management support.
- Copy services: Allows a user to execute copy services functions.

DS Command Line Interface

The IBM System Storage DS Command Line Interface (CLI) is a single utility that has the ability to perform a wide range of commands for both configuration and copy services activities. The CLI has the ability to dynamically invoke copy services functions. The DS CLI client is available for multiple operating system environments.

The DS8000 supports audit logging and log viewing of an exported log file. The DS8000 audit logging capability includes information such as a list of users who have logged in, when the user logged in, and what the user did during their session. A separate log entry is added each time a resource is created, deleted, or modified, providing enhanced administrator ease-of-use and additional security.

For more information about the DS8000 disk series please, refer to the following Web site:

http://www-03.ibm.com/systems/storage/disk/enterprise/index.html

XIV Storage System

To address the new requirements associated with next generation digital content, IBM has recently acquired a company called XIV and its Nextra™ architecture for its ability to scale dynamically, heal itself in the event of failure, and self-tune for optimum performance, all while eliminating the significant management burden typically associated with rapid growth environments. The architecture also is designed to automatically optimize resource utilization of all the components within the system, which can allow for easier management and configuration and improved performance and data availability. The Nextra architecture has been in production for more than two years, with more than four petabytes of capacity being used by customers today.

Steve Duplessie, Enterprise Strategy Group, says: *XIV is designed for Web 2.0 digital content that nobody had a clue about two years ago. This content requires outrageous levels of scale on demand, runs on cheap component-based hardware and is self-healing.*

The XIV architecture is well positioned for implementing storage solutions for so called emerging workloads such as Web 2.0, video. These emerging workloads are characterized by large data sets, unpredictable rapid growth, occasional high usage, and rapid data aging — that is, data is accessed frequently after creation and then becomes inactive.

The XIV architecture uses autonomic self-healing technology and is built around low cost off-the-shelf components. It offers massive parallelism while maintaining deployment and management simplicity. It uses optimized storage distribution algorithms to spread data as widely as possible and eliminate the probability of hotspots. It also offers thin provisioning, self healing functions and very fast internal replication with sub-second snapshot creation times and rapid rebuilds (one 750 GB drive is rebuilt in as little as 20 minutes).

XIV presents disk LUNs to servers, as with standard disk storage subsystems. It is a scale out architecture based on an on low cost interface nodes and storage nodes connected by an ethernet fabric, as illustrated in Figure 5-5.

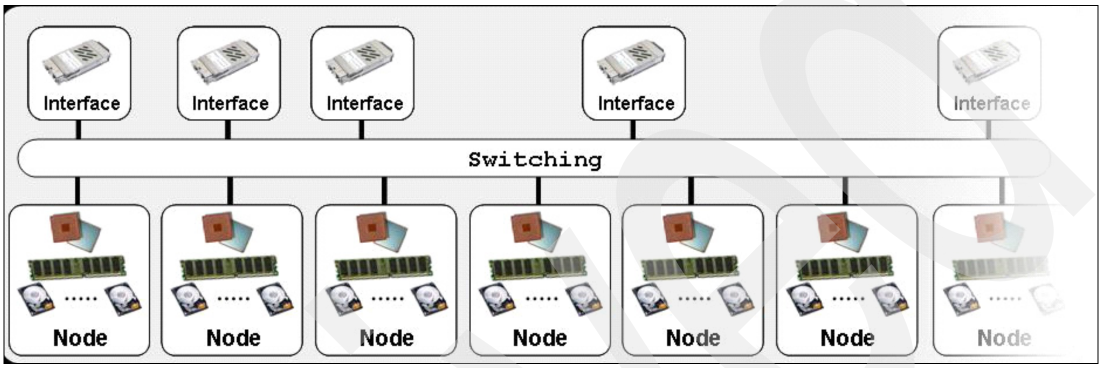


Figure 5-5 XIV architecture overview

Interface nodes receive I/O requests from servers, they analyze and decompose the I/O, and dispatch the resulting I/Os to storage nodes based on an internal request distribution algorithm. Storage and interface nodes connect to the internal switching network that offers both redundancy and any-to-any connectivity. The storage nodes contain cache memory and the actual disk devices to store the data.

Each virtual LUN as seen by the servers is spread across all drives and all nodes, the virtual LUN is cut into 1MB partitions and stored on the underlying disks inside the storage nodes. XIV's distribution algorithm automatically distributes partitions across all storage nodes in the system in a pseudo random way. All 1MB partitions are automatically replicated to two separate storage nodes for availability.

Data distribution only changes when the system changes, and equilibrium is kept when storage nodes are added, removed or there is a hardware failure. XIV offers space efficient point in time copies with sub-second creation time and thousands of point in time copies allowed per volume without performance degradation.

XIV offers a very simple management interface that completely hides the complexity of managing the storage system and facilitates operations such as creating, removing, reassigning, enlarging and copying storage LUNs.

From the information themes perspective XIV offers high performance and the ability to scale performance linearly, a very simple management interface to reduce management cost, internal copy functionality to create new LUNs, and is well positioned as an offline storage replacement and low cost storage solution.

5.2.2 Tape storage systems

Tape continues to figure significantly in IT infrastructures for high-capacity low-cost storage. Its unique attributes can help you manage storage growth requirements. Tape is:

- **Removable:** Store it away to help protect it from viruses, sabotage and other corruption
- **Scalable:** Add more cartridges, not drives
- **Portable:** Move it to another site to avoid destruction in the event of disaster
- **Fast:** Lightning fast, up to 120 MB per second, IBM LTO4 native data rate
- **Reliable:** IBM servo technology, read after write verification and advanced error correction systems, to help maintain reliable and dependable storage

Tape can help you address compliance requirements and WORM applications. Tape has a low total cost of ownership, costing up to 10 times less than disk. Using today's disk to disk to tape methodology, tape is a key element.

IBM offers a comprehensive range of tape products: tape drives, tape libraries, and virtual tape libraries, as illustrated in Figure 5-6.

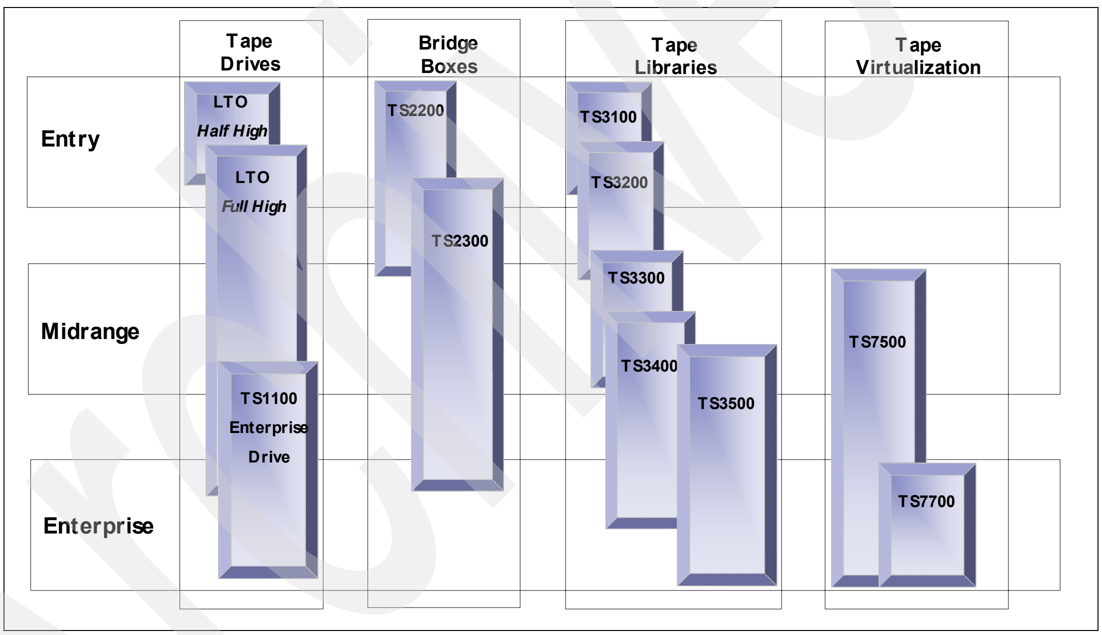


Figure 5-6 IBM System Storage tape portfolio

Tape drives

IBM offers two families of enterprise tape drives: the IBM TS1120 tape drive, and the IBM LTO family of drives, currently offering LTO4 technology.

TS1120 tape drive

The TS1120 tape drive offers a solution to address applications that require high capacity, fast access to data and long-term data retention. It is supported both in IBM tape libraries and as a standalone drive. The TS1120 tape drive uses IBM 3592 Cartridges, which are available in limited capacity (100 GB) for fast access to data, and standard capacity (500 GB) or extended capacity (700 GB) that help to reduce resources to lower total cost. All three cartridges are available in re-writable or Write Once Read Many (WORM) format. The TS1120 uses a linear serpentine recording technique and offers adaptive data rate technology to match the drive data rate to the requesting application's capabilities. The TS1120 includes data encryption capabilities within the drive itself, helping to avoid the necessity for host-based encryption of data.

TS1120 tape drives can be shared among supported open system hosts on a SAN. Sharing drives optimizes drive utilization and helps reduce infrastructure requirements. The TS1120 tape drive supports a native data transfer rate of up to 104 MBps. In open system environments where data typically compresses at 2:1, the TS1120 tape drive can transfer data up to 200 MBps. In a mainframe environment where data typically compresses at 3:1, a single tape drive can transfer data up to 260 MBps. This can help reduce data access and retrieval times and require fewer resources to support a given environment.

Linear Tape-Open

The Linear Tape-Open (LTO) program started as a joint initiative of IBM, Hewlett-Packard, and Seagate Technology. In 1997, the three companies set out to enable the development of best-of-breed tape storage products by consolidating state-of-the-art technologies from numerous sources. The LTO technology objective was to establish new open-format specifications for high-capacity, high-performance tape storage products. The three LTO sponsoring companies also took steps to protect client investment by providing a six-generation roadmap, shown in Figure 5-7, and establishing an infrastructure to enable compatibility between products. At the time of writing, four generations were available.

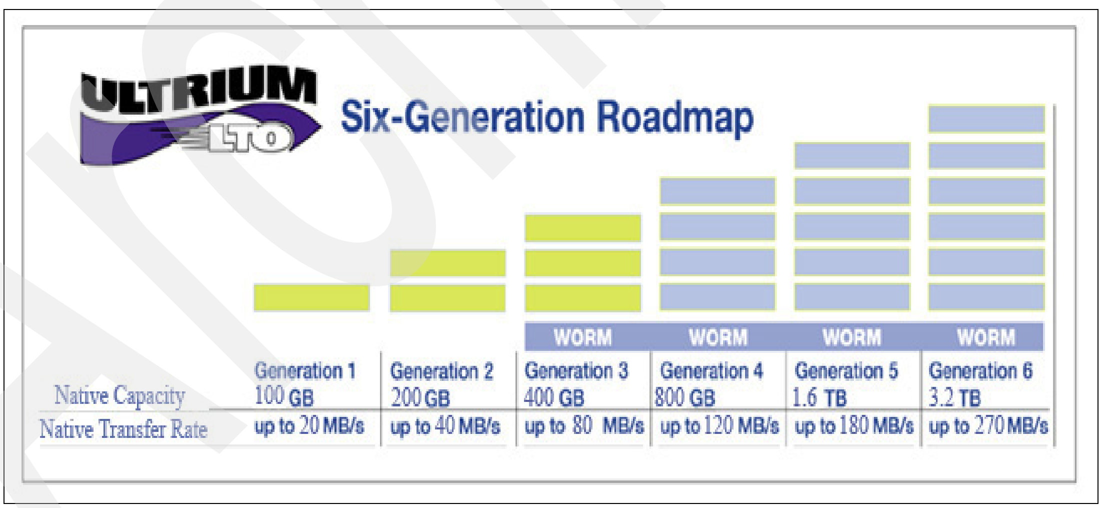


Figure 5-7 LTO Ultrium roadmap

The LTO Ultrium compatibility investment protection is provided based on the following principles:

- An Ultrium drive is expected to read data from a cartridge in its own generation and at least the two prior generations.
- An Ultrium drive is expected to write data to a cartridge in its own generation and to a cartridge from the immediately prior generation in the prior generation format.

Each LTO generation consists of new drives and new cartridge media. Figure 5-8 shows the compatibility between multiple LTO generation drives and media.

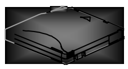
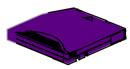
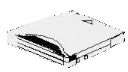
Generation	Native Capacity	Data rate	Gen 1	Gen 2	Gen 3	Gen 3 HH	Gen 4
	100 GB	Read	15 MB/sec				
		Write	15 MB/sec				
	200 GB	Read	15 MB/sec+	35 MB/sec			
		Write					
	400 GB	Read	15 MB/sec+	35 MB/sec+	80 MB/sec	60 MB/sec	
		Write					
	800 GB	Read		35 MB/sec+	80 MB/sec+	80 MB/sec+	120 MB/sec
		Write					120 MB/sec

Figure 5-8 LTO Cartridge Interchangeability

IBM currently offers the TS1030 LTO3 and TS1040 LTO4 tape drives. The IBM System Storage TS1040 LTO tape drive, also referred to as LTO4, offers high capacity, performance, and technology designed for the midrange environment. The TS1040 LTO4 offers either a 4 Gbit Fibre Channel interface for either point-to-point or Fibre Channel-Arbitrated Loop attachment, a Ultra160 LVD SCSI attachment, or a 3 Gbps dual-ported SAS interface.

The TS1040 is the first tape drive within the IBM LTO tape family that supports Tape Encryption. Tape Encryption is supported on the SAS and Fibre Channel tape drives. The TS1040 offers 800 GB native physical capacity (1.6 TB compressed at 2:1) on LTO Ultrium 4 media and up to 120 MB/s native data transfer rate. The TS1040 offers digital speed matching (30, 48, 66, 84 103, 120 MB/s) and has a 256 MB Internal Buffer (128 MB for IBM Ultrium 3). Several continued features/functions from IBM Ultrium 3 are:

- WORM technology
- Dual stage 16-channel head actuator
- Independent tape loader and threader motors
- Graceful dynamic braking
- SARS (Statistical Analysis and Reporting System) and ECC (Error Correction Code)
- Same 5 ¼” form factor

Encryption Key Manager

The IBM Encryption Key Manager component is Java™ based and can help generate and manage encryption keys for TS1120 and TS1040 tape drives across the enterprise. This feature uses standard key repositories on supported platforms and supports three different encryption management methods: application managed, system managed, and library managed. The TS1120 and TS1040 tape drives supports transparent encryption, minimizing application changes in the system and library managed implementations. The TS1120 and TS1040 tape drive encryption capability is designed to avoid the necessity for application changes in the system and library managed implementations.

For additional information on IBM tape drives, refer to:

http://www-03.ibm.com/systems/storage/tape/index.html

Tape libraries

IBM System Storage provides a complete spectrum of tape libraries highlighting high performance and capacity for entry, midrange, and enterprise system scenery. The IBM libraries handle backups, save and restore, and archival data storage requirements. There are many tape library models to choose from, as shown in Figure 5-9. We illustrate some examples, but we concentrate on features and capabilities of the larger ones.

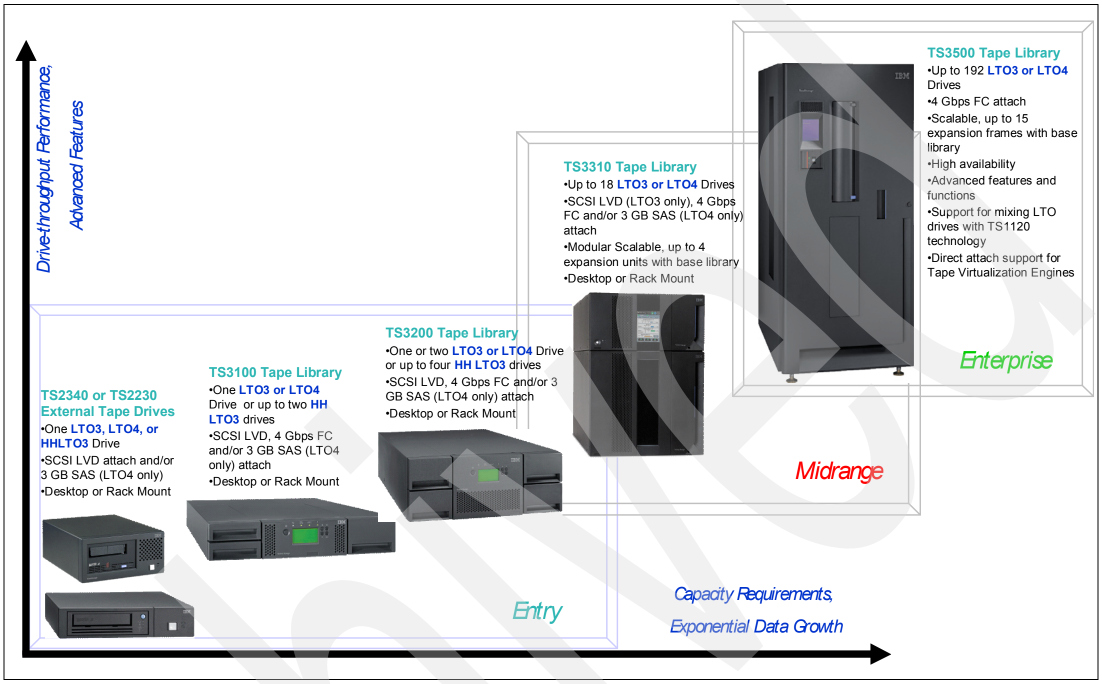


Figure 5-9 A sample set of IBM tape libraries

TS3100 Tape Library

The IBM System Storage TS3100 Tape Library is well-suited for handling backup, save and restore, and archival data-storage requirements for small to medium-size environments. It has a single Ultrium 4 tape drive and 22 tape-cartridge capacity.

TS3200 Tape Library

The IBM System Storage TS3200 Tape Library is designed for backup, save and restore, and archival data-storage requirements for small to medium-size environments. The TS3200 is an external 4U standalone or rack-mountable unit that incorporates up to two Ultrium 4 tape drives and 44 tape cartridges, as well as 3 media mail slots and 1 dedicated cleaning cartridge slot.

TS3310 Tape Library

IBM System Storage TS3310 Tape Library offers simple, rapid expansion as processing requirements grow. Its entry level configuration is a single 5 EIA rack unit high library. Over time, as the necessity for tape backup expands, you can add an additional 9U expansion module, with space for additional cartridges, tape drives and a redundant power supply.

The TS3310 is available in a base frame and multiple expansion frames that offers up to 18 IBM LTO3 or LTO4 tape drives. The base frame (L5B) has up to 2 drive capacity and 36 cartridge slots and up to four expansion frames (E9U) can be added, each with up to 4 drives and 92 cartridge slots giving a total library capacity of 402 slots and up to 633.6 TB with LTO4 media (with 2:1 compression). The TS3310 can be partitioned and divided into one logical library per installed tape drive. These logical libraries can be simultaneously connected to a wide variety of different servers, running different operating systems and tape applications.

TS3500 Tape Library

The IBM System Storage TS3500 Tape Library combines IBM automation and drive technology to provide a highly scalable, automated tape library for System z™ and open systems backup and archive in midrange to enterprise environments.

The IBM System Storage TS3500 Tape Library is a highly scalable, automated tape library for mainframe and open systems backup and archive in midrange to enterprise environments. It is formerly known as IBM TotalStorage Tape Library 3584. The TS3500 supports System z nd open systems. It can accommodate 3592 J1A, TS1120, TS1030 and TS1040 LTO tape drives. The TS3500 Model HA1 allows two robotic accessors to operate simultaneously in configurations from two to 16 frames. The configuration highlights are:

- 1 - 16 frames (plus two service frames)
- 1 - 192 tape drives
- TS1030 TS1040 LTO or TS1120/3592 J1A tape drives are supported

  Tape drives types and associated media require unique frames
- Designed for concurrent maintenance
  - Hot-swappable drives
  - Hot-swappable drive and accessor power supplies
  - Redundant drive and accessor power
  - Redundant AC power
- Dual Ethernet interface for TSSC (TS3500 System Console) connection
  - High Availability Library Management / Robotics
  - Adds left and right service frames to the library
  - Adds second active accessor and library management node
  - Each accessor operates independently in flexible zones
  - Support dynamic workload balancing across zones
- IBM System z and z/OS options
  - Supports redundant ESCON/FICON attachment to supported servers
  - Redundant 3953 F05 frame power and library control path interfaces
  - Redundant 3953 F05 VTS switches features (optional for TS1120 controller)
  - Optional redundant 3953 Library Manager

The TS3500 Advanced Library Management System (ALMS), virtualizes the locations of cartridges in the TS3500. Logical libraries can then consist of unique drives and ranges of volume serial numbers instead of fixed locations. With ALMS, the TS3500 is the industry's first standards-based tape library to virtualize the locations of cartridges (called SCSI element addresses) while maintaining native SAN attachment for the tape drives. ALMS enables logical libraries to consist of unique drives and ranges of volume serial (VOLSER) numbers, instead of fixed locations.

When you enable ALMS with its license key, you can assign tape drives to any logical library by using the Tape Library Specialist Web interface. Logical libraries can also be added, deleted, or easily changed without disruption. Storage capacity can be changed without impact to host applications. ALMS offers dynamic management of cartridges, cartridge storage slots, tape drives, and logical libraries. It enables the TS3500 to achieve unprecedented levels of integration for functionality through dynamic partitioning, storage slot pooling, and flexible drive assignment.

ALMS eliminates downtime when you add Capacity on Demand (CoD) storage, add or remove logical libraries, or change logical library storage allocation. ALMS also reduces downtime when you add expansion frames, add or remove tape drives, or change logical drive allocation.

The capabilities of ALMS include:

- Dynamic partitioning (storage slot pooling and flexible drive assignment)
- The transparent ability to add or remove storage capacity to any host application
- The ability to configure drives or to configure Model L22, L23, L32, L53 and L52 storage capacity without taking the library offline
- Cartridge assignment policy

3494 Tape Library

The IBM TotalStorage 3494 Tape Library is an excellent solution for today's large storage requirements. Modular, flexible, and reliable, the 3494 uses 3590 tape drives, is cost-effective, and is backed by the service expertise of the IBM service organization. The 3494 offers a variety of models and features to fit all your requirements and will grow as your business grows.

For additional information on IBM tape libraries, refer to:

`http://www-03.ibm.com/systems/storage/tape/index.html`

Virtual tape libraries

IBM offers two families of tape virtualization products - the TS7500 and the TS7700.

TS7700

The IBM Virtualization Engine™ TS7700 is a mainframe virtual-tape solution that is designed to optimize tape processing. Through the implementation of a fully integrated tiered storage hierarchy of disk and tape, the benefits of disk and tape technologies can be leveraged to help enhance performance and provide the capacity required for today's tape processing requirements. Deploying this innovative subsystem can help reduce batch processing time, total cost of ownership and management overhead.

TS7520

The IBM Virtualization Engine TS7520 combines hardware and software into an integrated tiered solution designed to provide tape virtualization for open systems servers connecting over Fibre Channel and iSCSI physical connections. When combined with physical tape resources for longer term data storage, the TS7520 Virtualization Engine is designed to provide an increased level of operational simplicity and energy efficiency, support a low cost of ownership and increase reliability to provide significant operational efficiencies. With backup windows shrinking, tolerance for hardware failure has virtually disappeared. The TS7520 Virtualization Engine is designed to help address these issues by reducing tape mechanical delays and providing fault tolerant architecture options supporting high availability.

The TS7520 supports the following tape operating modes:

- Virtual, emulation, support of IBM LTO-2, LTO-3 and LTO-4* Tape Drives, and TS1120 Tape Drives Model J1A and E05
- Virtual, emulation, support of an IBM TS3500 Tape Library
- Virtual, emulation, support of the IBM TS3310, TS3200, TS3100 Tape Libraries
- Physical backend direct attach TS1120 Model J1A/E05, LTO-2, LTO-3, LTO-4*
- Physical backend direct attach support for TS3500, TS3310, TS3200, TS32100 and 3494 Tape Libraries

The TS7520 is built with DS4200 disk storage IBM System x™ Servers and tape virtualization software. The TS7520 offers:

- Up to 884 TB capacity w/500 GB SATA drives
- Up to1.3 PB capacity w/750 GB SATA drives (Preview)
- Up to 4.8 GB/sec data transfer performance
- Up to 24 Fibre Channel support
- Up to 512 virtual libraries
- Up to 4096 virtual drives
- Up to 256,000 virtual cartridges
- Active failover
- Enhanced caching provides the backup application direct access to data whether in cache or on physical tape and a more intelligent policy based data migration
- Encryption on network or backstore tape to help protect sensitive customer information
- HW assisted compression designed to improve system performance with replication, compression and encryption
- NDMP, which enables backup applications and NAS devices to perform backup and restore using the NDMP version 4 protocol over an IP network
- Control path failover and data path failover, which can help provide higher availability over the control path
- Path Failover enables automatic control and data path failover to a pre-configured redundant path without aborting the current job in progress
- Failover/Failback, which enables hardware connections between two installed TS7520 Virtualization Engines in the same 3952 Tape Frame
- iSCSI, which provides connectivity to IP connected hosts
- Secured Tape, allowing a mechanism to encrypt the data when exported to physical tape
- Hosted backup, allowing you to run supported backup applications on the TS7520

TS7520 disaster recovery

The TS7520 offers disaster recovery functionality by using the network replication feature. It provides a method to recover from complete data loss by sending copies of data offsite. There are three methods of Network Replication: Remote Copy, Replication, and Export Network Copy. To use the Network Replication function, you require two IBM Virtualization Engine TS7520s:

- The local TS7520 that serves virtual tape drives to your backup servers
- A disaster recovery/remote TS7520

Remote Copy is a manually triggered, one-time replication of a local virtual tape. Upon completion of the Remote Copy, the tape resides on the primary TS7520 and in either the Virtual Vault or a virtual library on the remote TS7520.

The Replication process is either triggered by a scheduled event or when the new or changed data on virtual volume reaches a certain predetermined size. When Replication is configured, a local virtual volume is created and linked to the virtual replica on the remote Virtualization Engine. A replica tape is always linked to the original virtual tape. The replica tape receives incremental changes from the local source tape, ensuring the two tapes are always in-sync at the end of a replication session.

Export Network Copy involves a one-time copy or move of the virtual tape when the backup software has sent an export command (backup application command that issues a SCSI Move Medium to I/O Slot). Export Network Copy provides for one-time copy or move after the eject, but the destination is the remote TS7520 Virtualization Engine.

For more information on the IBM TS7520, refer to:

http://www.redbooks.ibm.com/abstracts/sg247520.html?Open

5.2.3 N series Unified Storage

The IBM Storage System N series provides a range of reliable, scalable storage solutions for a variety of storage requirements. These capabilities are achieved by using network access protocols such as NFS, CIFS, HTTP and iSCSI as well as SAN technologies such as Fibre Channel. Utilizing built-in RAID technologies all data is well protected with options to add additional protection through mirroring, replication, snapshots and backup. These storage systems are also characterized by simple management interfaces that make installation, administrating and troubleshooting uncomplicated and straightforward. The IBM System Storage N series is designed from the ground up as a standalone storage system.

Figure 5-10 illustrates a comprehensive range of models and two N Series model types:

- The A models contain internal storage.
- The G models are gateways and are connected to external storage.

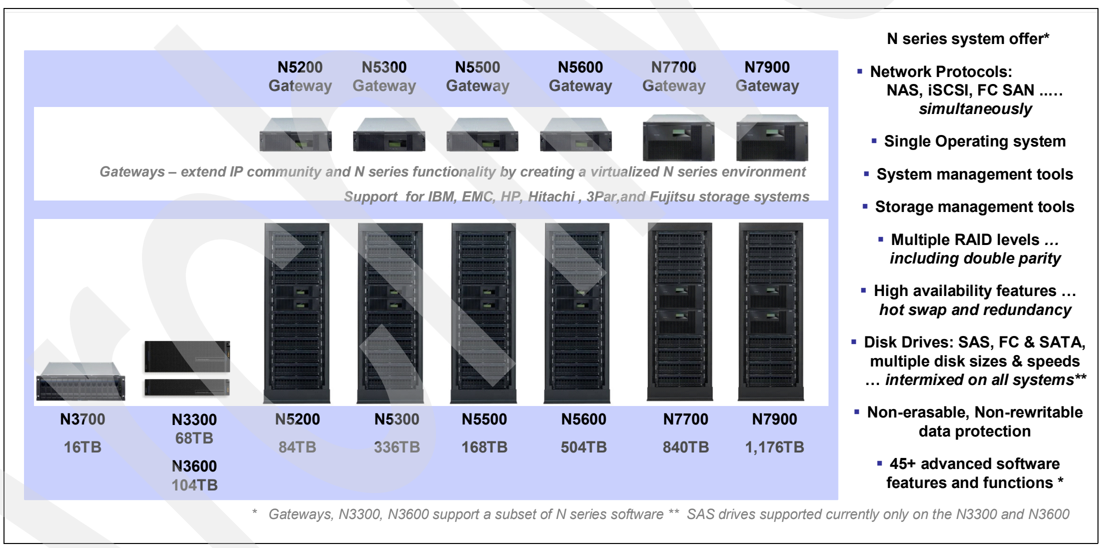


Figure 5-10 The range of IBM N Series models

The IBM N series A model storage systems offer multiprotocol connectivity using internal storage or storage provided by expansion units as shown in Figure 5-11.

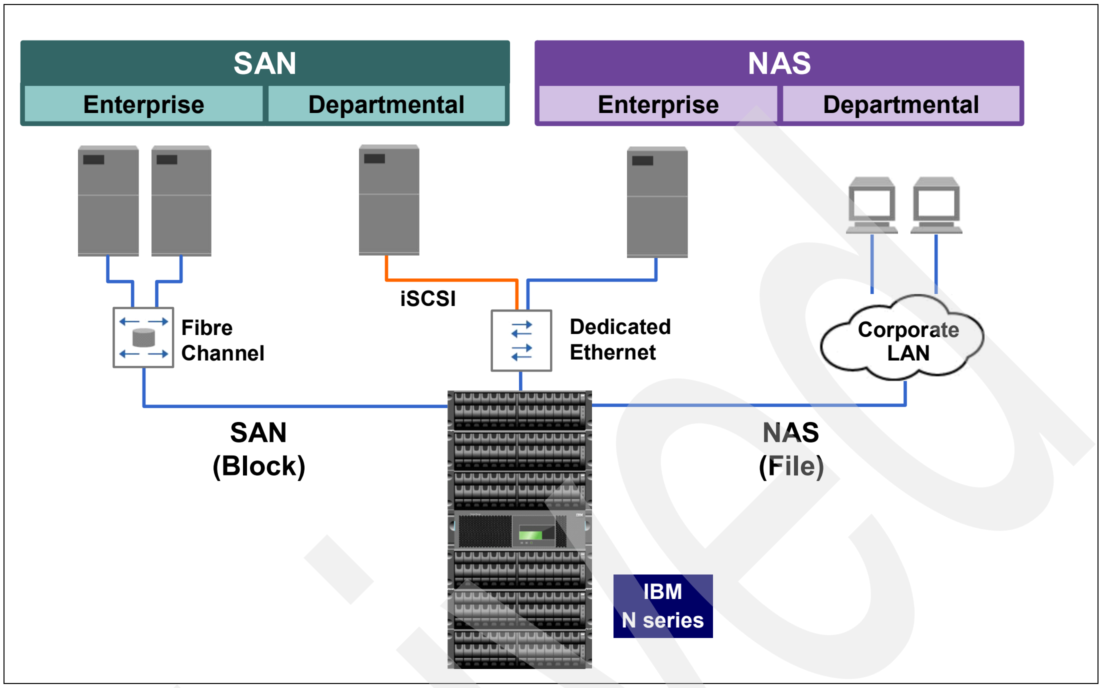


Figure 5-11 IBM N Series network connectivity options

The IBM System Storage N series systems are designed to provide integrated block- and file-level data access, allowing concurrent operation in IP SAN (iSCSI), FC SAN, NFS and CIFS environments.

The IBM N series models are a specialized, "thin server" storage system with a customized operating system, similar to a stripped down UNIX kernel, called to as Data ONTAP®. With a reduced operating system, the objective is to improve performance and reduce costs by eliminating unnecessary functions normally found in the standard Operating Systems.

The IBM N series come with pre-configured software and hardware, and with no monitor or keyboard for user access. This is commonly termed a "headless" system. A storage administrator accesses the systems and manages the disk resources from a remote console using a Web browser or command line.

One of the typical characteristics of a IBM N series storage systems product is its ability to be installed rapidly using minimal time and effort to configure the system. It can be integrated seamlessly into the network.

The IBM N series Gateway or G models will present shares, exports or LUNs that are built on external storage managed by the Gateway. N series Gateways take storage array LUNs (which are treated as disks) and virtualize them through Data ONTAP, presenting a unified management interface, as illustrated in Figure 5-12.

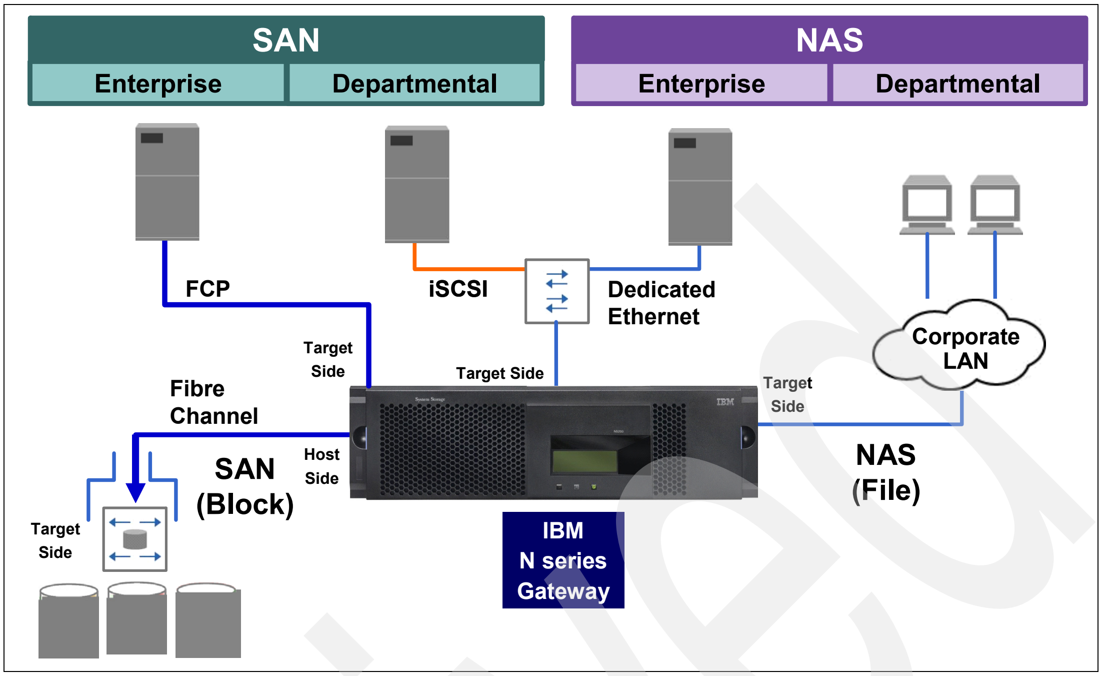


Figure 5-12 IBM N Series Gateway models

The N Series Gateways allow you to reutilize existing storage assets.

N Series systems offer a comprehensive set of software functions for many storage and storage management challenges. Some examples of the functions that are offered are:

- Network Access Protocols:
  - CIFS file system protocol over IP for Windows clients
  - NFS file system protocol over IP for UNIX/Linux® clients
  - iSCSI for Windows/UNIX/Linux server attached to storage via IP network utilizing block I/O protocols
  - FCP for Windows/UNIX/Linux server attached to storage via fiber channel network utilizing block protocols
- Data protection and Business Continuity:
  - On board copy services via Snapshot™ and SnapRestore – virtual file & complete volume copy
  - Outboard copy services via SnapVault®, SnapMirror, SyncMirror® to send data to a second remote N Series system
  - Double parity RAID provides enhanced data protection for SATA drives
  - Cluster Failover – between redundant N Series nodes
  - SnapLock and LockVault™ for non-erasable non-rewritable WORM data protection
- System tools, usability aids, provisioning:
  - FilerView® for overall system monitoring and management
  - SnapManager® for snapshot backups of Exchange and SQL environments
  - SnapDrive® – usability for block I/O environments
  - FlexClone® for database cloning
  - FlexVol® for thin provisioning
  - MultiStore® for N Series partitioning
  - A-SIS deduplication support for single instance store of multiple copies of the same data, saving storage space

- Specific solutions for business problems:
  - E-mail archive
  - Microsoft Exchange, SQL, and Oracle consolidations
  - Storage consolidation
  - Server consolidation
  - Catia migrations
  - Unified storage
  - Corporate Compliancy
  - Information Lifecycle Management
  - Infrastructure Simplification

For more information on the N Series family, refer to:

http://w3.itso.ibm.com/abstracts/sg247129.html?Open

5.2.4 Block virtualization with SAN Volume Controller

The IBM System Storage SAN Volume Controller (SVC) enables a single point of control for diverse, heterogeneous storage resources to help drive improved application availability and greater resource utilization. The objective is to identify all storage resources in your IT infrastructure and to make sure they are used as efficiently as possible, while avoiding administrative cost.

SAN Volume Controller is designed to pool storage volumes from IBM and non-IBM storage systems into a single reservoir of capacity for centralized management. SAN Volume Controller is designed to hide the boundaries among disk systems, which helps you to focus on managing storage as a resource to meet business requirements and not as a set of boxes. SAN Volume Controller helps you to set business process goals based on all the storage resources at your disposal rather than allowing the storage resources to dictate what your business can accomplish.

Figure 5-13 illustrates the high level architecture of the SAN Volume Controller.

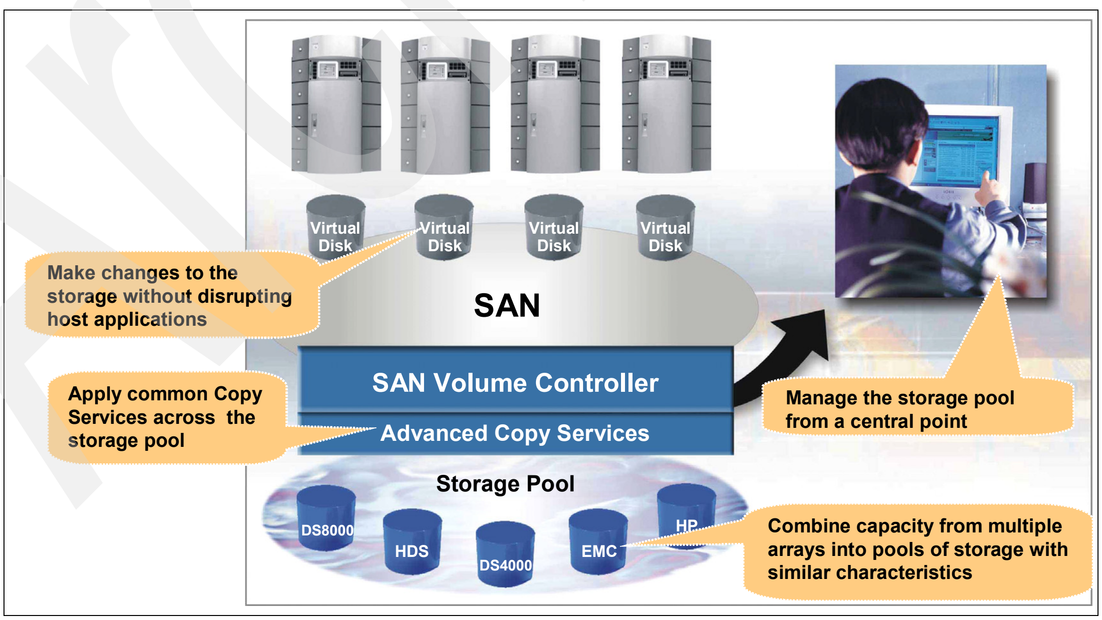


Figure 5-13 SAN Volume Controller architecture

SAN Volume Controller is highly scalable. A SAN Volume Controller is composed of *I/O Groups*. An I/O Group is made by combining two System x servers, each server includes a four-port 4 Gbps-capable host bus adapter (HBA). Each I/O Group contains 8 GB of mirrored cache memory. Highly available I/O Groups are the basic configuration element of a SAN Volume Controller cluster. Adding I/O Groups to the cluster increases cluster performance and bandwidth.

An entry-level SAN Volume Controller configuration contains a single I/O Group. SAN Volume Controller can scale out to support four I/O Groups, and it can scale up to support 1024 host servers. For every cluster, SAN Volume Controller supports up to 4096 virtual disks. This configuration flexibility means that SAN Volume Controller configurations can start small with an attractive price to suit smaller customers or pilot projects and yet can grow to manage very large storage environments. SAN Volume Controller Version 4.2.1 adds the ability to support up to eight petabytes of storage, four times the previous limit, enabling the benefits of SAN Volume Controller to be experienced in even larger configurations.

SAN Volume Controller provides a graphical user interface (GUI) for central management. Using this interface, administrators can perform configuration, management and service tasks in a consistent manner over multiple storage systems even from different vendors. SAN Volume Controller allows administrators to map disk storage volumes to virtual pooled volumes to help use you existing storage more efficiently. SAN Volume Controller also incorporates the Storage Management Initiative Specification (SMI-S) application programming interface (API). SAN Volume Controller Version 4.2.1 uses the IBM System Storage Productivity Center (SSPC), an advanced management console that can provide a view of both IBM- and non-IBM storage environments.

Replication services

SAN Volume Controller offers a comprehensive set of replication services. With many conventional SAN disk arrays, replication operations are limited to in-box or like-box-to-like-box circumstances. But SAN Volume Controller enables administrators to apply a single set of advanced network-based replication services, such as FlashCopy to make instantaneous copies, across multiple storage systems from different vendors. This ability can help simplify the storage environment and reduce the total cost of storage.

Network-based SAN Volume Controller replication services can help match the value of data with the cost of storage. For example, whereas production data can be stored on enterprise-class storage, SVC is designed to enable backup copies created with the FlashCopy function to be stored on lower cost storage. Similarly, conventional approaches to business continuity require largely the same storage at production and recovery locations. But SVC is designed to support different storage at each location, which can help to reduce cost when creating disaster recovery strategies.

SAN Volume Controller supports a broad range of replication services that operate in a consistent manner regardless of the type of storage being used with SVC. The FlashCopy function is designed to create an almost “instant” copy of active data, which could be used for backup purposes or for parallel processing activities. Up to sixteen copies of data can be created. Metro Mirror and Global Mirror operate between SVC systems at different locations to help create copies of data for use in the event of a catastrophic event at a data center. Metro Mirror is designed to maintain a fully synchronized copy at “metropolitan” distances (up to 300 km) whereas Global Mirror is designed to operate asynchronously and so helps maintain a copy at much greater distances (up to 8000 km).

SVC 4.2.1 replication services

SAN Volume Controller Version 4.2.1 adds important new enhancements to replication services:

- Incremental FlashCopy provides the ability to copy only the portions of the source or target virtual disk (VDisk) that have been updated since the FlashCopy function was last used, instead of copying the whole VDisk. This capability helps enable the FlashCopy function to complete more quickly, thus reducing the impact of using FlashCopy.
- Cascaded FlashCopy provides the ability to initiate a copy from a Vdisk that is the target of another FlashCopy relationship, which provides even greater flexibility in using the FlashCopy function. The total number of targets from the original source, or based on intermediate target sources, can be up to sixteen.
- Dynamically Configurable Replication Services Space enables users to configure up to 256 TB of replication services capacity per I/O Group as required, allocatable across FlashCopy, Metro Mirror, and Global Mirror as required. The new capability enables greater flexibility in deploying replication services to better meet the requirements of different installations.

Summary

Because it hides the physical characteristics of storage from host systems, SAN Volume Controller is designed to help insulate host applications from physical changes to the storage pool. This ability can help enable applications to continue to run without disruption while you make changes to your storage infrastructure, which can help your business increase its availability to its customers.

In many IT environments, inactive data can make up the bulk of stored data. SAN Volume Controller allows administrators to control storage growth more effectively by moving low-activity or inactive data into a hierarchy of lower-cost storage. SAN Volume Controller makes it easier to implement tiered storage, which allows a mix of different types of storage to be used, to reduce overall costs and support movement of data between tiers without disruption to applications. As the SVC also has cache, it can improve the performance of lower tier storage, enabling it to be used more widely in a data center, further reducing costs.

For more information on the SAN Volume Controller, refer to:

http://www-03.ibm.com/systems/storage/software/virtualization/svc/index.html

5.2.5 Tivoli Storage Manager, SSAM, and DR550 data management products

IBMTivoli Storage Manager (TSM) family offerings are designed to provide centralized, automated data protection and data management solutions. From the ECM perspective, Tivoli Storage Manager offers a very powerful archival management platform, as it can store and retain data objects and manage them based on predefined policies. The ECM application can hand the data to Tivoli Storage Manager, which will perform all the necessary storage management operations.

Tivoli Storage Manager “Inactive data management” has many meanings and applies to many types of data. It can be files that are not being used as frequently as they once were. It can be bookkeeping or regulatory compliance data that has to be stored for many years. It can also be copies of active data that are being kept for recovery purposes. Regardless of what type of data it is, two things are true – inactive data is what is driving much of the capacity growth requirement in IT environments today, and there are a lot of questions on how to manage the various types of data.

IBM Tivoli Storage Manager has many types of data management techniques; the same product can be used to perform backup operations, hierarchical space management (HSM), archive, and data retention. Tivoli Storage Manager is a data management platform.

Tivoli Storage Manager is a client-server software application that can run on many platforms and operating systems. Tivoli Storage Manager has also special versions that offer protected or WORM storage and can come pre-integrated into a secured hardware appliance. In this section we will discuss:

- **Tivoli Storage Manager:** The software product and the functionality it offers
- **System Storage Archive Manager (SSAM):** A software “hardened” version of Tivoli Storage Manager
- **DR550:** A secured hardware appliance built around IBM System Storage devices and SSAM

Tivoli Storage Manager

Tivoli Storage Manager provides a comprehensive set of services focused on the key data protection and management tasks of backup, archive, recovery, space management, and disaster recovery.

Tivoli Storage Manager allows you to separate the backup, archiving, and retention of data from storage-related management aspects of the data, in addition to many other services. Tivoli Storage Manager offers various storage management functions relevant to ECM systems:

- Data archiving defines how to insert data into the data retention system. Tivoli Storage Manager offers a command line interface to archive and back up files and a C language application programming interface (API) for use by content management applications.
- Data retention defines how long to keep the data object, not the individual tape. Tivoli Storage Manager offers various data retention options, such as:
  - By date specifies the duration to retain the data.
  - Event-based determines retention on notification of a future event.
  - Deletion hold prevents deleting an object even after its defined retention period.
- Storage defines on which storage device to put the object. Tivoli Storage Manager supports hundreds of disk and tape storage devices and integrated hierarchical storage management of stored data. You can choose the most effective storage device for your requirements and subsequently let the data automatically migrate to different storage tiers.
- Non-erasable, Non-rewritable (NERE) functionality is offered by System Storage Archive Manager. The Tivoli Storage Manager administrator cannot accidentally or intentionally delete objects stored in Tivoli Storage Manager.
- Storage management services are provided by Tivoli Storage Manager. These additional storage management services facilitate hardware replacement and disaster recovery. Tivoli Storage Manager allows for easy migration to new storage devices when the old storage devices require replacing, and this is likely to happen when data is retained for long periods of time. Tivoli Storage Manager also offers functions to make multiple copies of archived data.

Tivoli Storage Manager offers a strong and comprehensive set of functions that you can exploit to effectively manage archived data. You can consider Tivoli Storage Manager an abstraction or virtualization layer between applications requiring data retention or storage management services and the underlying storage infrastructure.

Tivoli Storage Manager components

Tivoli Storage Manager is a client server software application that provides services such as network backup and archive of data to a central server. It has two main functional components:

- The Tivoli Storage Manager client is a component that you install on systems that require Tivoli Storage Manager services. The Tivoli Storage Manager client accesses the data to be backed up or archived and is responsible for sending the data to the server.
- The Tivoli Storage Manager server is the central repository for storing and managing the data received from the Tivoli Storage Manager clients. The server receives the data from the client over the LAN network, inventories the data in its own database, and stores it on storage media according to predefined policies.

Storing objects

Tivoli Storage Manager stores object metadata describing stored objects, policies and storage locations for each object into a powerful relational database. The flexibility of the Tivoli Storage Manager database enables you to define storage management policies around business requirements for individual clients or groups of clients. You can assign client data attributes, such as the storage destination, number of versions, and retention period at the individual file level.

Tivoli Storage Manager Object data is stored in Tivoli Storage Manager storage pools. Tivoli Storage Manager performs multiple diverse hierarchy and storage media management functions by moving or copying data between different pools or tiers of storage, as shown in Figure 5-14.

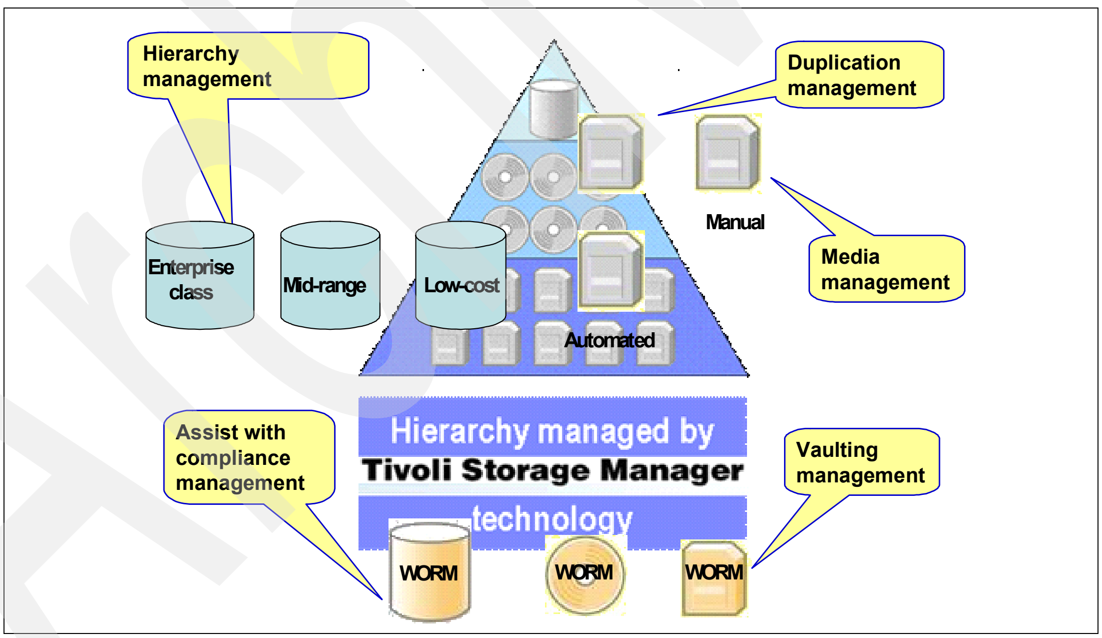


Figure 5-14 Tivoli Storage Manager supported storage devices

A Tivoli Storage Manager server can write data to more than 400 types of devices, including hard disk drives, disk arrays and subsystems, standalone tape drives, tape libraries, and other forms of random and sequential-access storage. The Tivoli Storage Manager server stores data objects on media grouped into Tivoli Storage Manager storage pools. You can connect external storage devices directly to the server through SCSI, through directly attached Fibre Channel, or over a Storage Area Network (SAN).

Media management capabilities

Tivoli Storage Manager provides sophisticated media management capabilities that automate and simplify the following tasks:

- Track multiple versions of data objects
- Move data objects automatically to the most cost-effective storage media
- Maintain multiple copies of critical data objects on separate storage devices and device types, for example a tape copy of data on disk.
- If a primary storage volume is damaged, Tivoli Storage Manager will automatically switch to using a secondary copy of the data
- Periodically create and maintain offsite copies of data
- Recover a complete Tivoli Storage Manager environment using the offsite copies
- Seamlessly perform data migration from one storage technology to another
- Support protected storage NERE devices
- Reutilize partially filled tape volumes by consolidating valid data onto new volumes and freeing old volumes

ECM application interaction

An interesting use of the Tivoli Storage Manager storage hierarchy, where data is stored into Tivoli Storage Manager from an ECM application, is shown in Figure 5-15. Based on policies set up by the system administrator, the data will be stored on the primary storage pool disk for fast access and retrieval when it is received from the ECM application. Tivoli Storage Manager monitors object age and pool utilization. When the disk storage pool space exceeds predefined utilization thresholds or when the object exceeds a pre-defined age, the object will be migrated automatically to the next storage pool, tape in our example.

Tivoli Storage Manager also allows for multiple copies of data objects to be created either synchronously or asynchronously by scheduled process to ensure that objects archived to Tivoli Storage Manager are protected against primary storage pool media failure. In the event of primary media failure, Tivoli Storage Manager will retrieve the object from the copy pool.

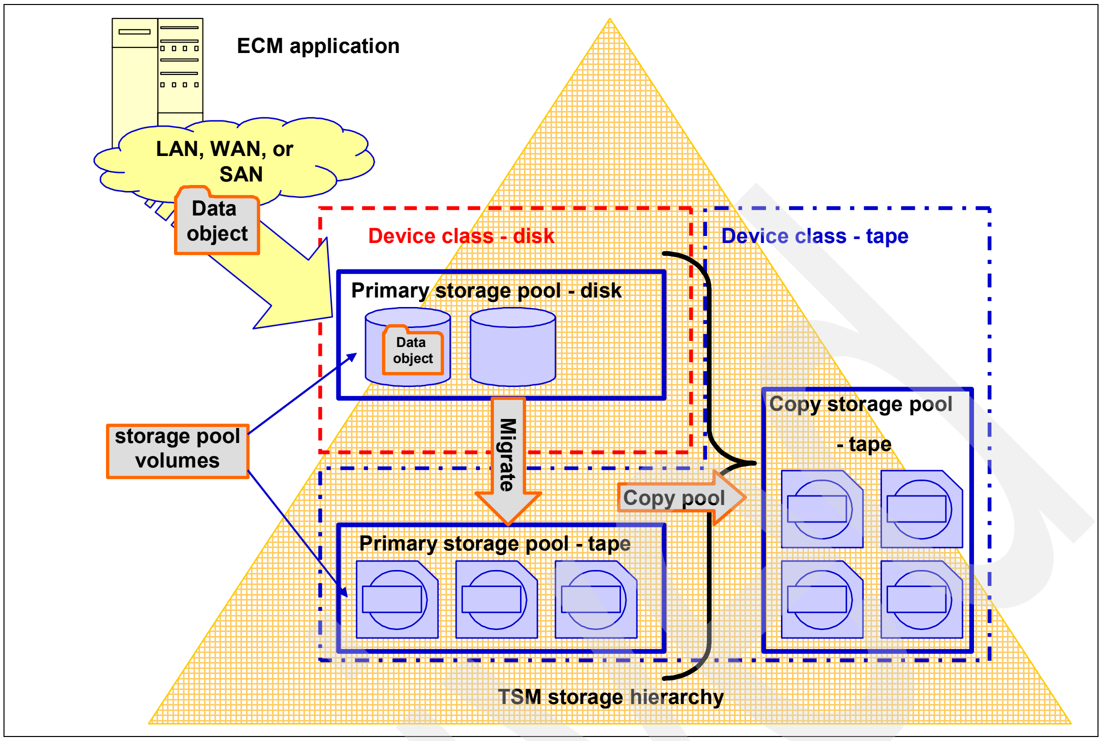


Figure 5-15 Tivoli Storage Manager storage hierarchy

Client-server program

Tivoli Storage Manager is a client-server program. You must install the client product on the system where you want to obtain Tivoli Storage Manager services. The client portion is responsible for sending and receiving data to and from the Tivoli Storage Manager server. The Backup Archive client has two distinct features:

- The backup component allows users to back up a number of versions of their data onto the Tivoli Storage Manager server and to restore from these, if the original files are lost or damaged. Examples of loss or damage are hardware failure, theft of computer system, or virus attack.
- The archive component allows users to keep a copy of their data for long term storage and to retrieve the data if necessary. ECM systems most often interact with Tivoli Storage Manager using the archive feature. Examples of this are to meet legal requirements, to return to a previous working copy if the software development of a program is unsuccessful, or to archive files that are not currently necessary on a workstation.

Interfaces

Tivoli Storage Manager also offers an API client to allow applications to access and store data directly in Tivoli Storage Manager. You can interact with the Tivoli Storage Manager server to run a backup/restore or archive/retrieve operation through three different interfaces:

- Graphical User Interface (GUI)
- Command Line Interface (CLI)
- Web Client Interface (Web Client)
- Application Programming Interface (API)

Data encryption

Tivoli Storage Manager offers data encryption can provide enhanced security for businesses via 128 bit AES or 56 bit DES encryption technology. Encryption options allow Tivoli Storage Manager to manage encryption keys (key management for each object) transparent to the application, or allow an application to manage encryption keys externally to Tivoli Storage Manager. The application stores and uses the keys to retrieve data. Encryption is enabled or disabled through an option in the Tivoli Storage Manager client.

> **Note:** Tivoli Storage Manager preview announcements have been made for support of data deduplication in the Tivoli Storage Manager server. Data deduplication offers the potential for substantial storage savings when multiple copies of the same data are stored on Tivoli Storage Manager managed disk storage. For additional information, refer to:
>
> http://www-1.ibm.com/support/docview.wss?rs=1018&context=SSSQZW&context=SSGSG7&dc=D600&uid=swg21295554&loc=en_US&cs=UTF-8&lang=en

Tivoli Storage Manager Hierarchical Storage Management (HSM)

Tivoli Storage Manager offers two separate space management clients for file systems: one for UNIX and one for Windows environments. In both cases, the space management client resides on the file server where you want to perform space management. It moves files from the local file system to lower cost storage managed by the Tivoli Storage Manager server, and this movement is called migration. Tivoli Storage Manager performs this movement based on administrator-defined criteria such as file size and age.

Moving a file to the Tivoli Storage Manager server implies that the file is removed from the Tivoli Storage Manager client. The client file system continues to see the file as though it were still on local disk. When a request to access the file occurs, the space management client intercepts the file system requests and, depending on operating system platform, either recalls the file to primary storage or, in some cases, can redirect the file system request to secondary storage.

These operations are performed transparently to the file system request even though the request can be slightly delayed because of the tape mount processing. An example of space management processing is shown in Figure 5-16. At the top data resides in a file system. The file system is monitored for space utilization and when this exceeds a predefined threshold the HSM migration function is invoked. The migration function will determine migration candidates based on attributes such as file size and age and start migrating them to the Tivoli Storage Manager server, and removing them from the file system leaving only a pointer or stub file.

The Tivoli Storage Manager server stores the objects on a predefined storage pool that uses SATA disk in the example. When a request is made for a migrated file Tivoli Storage Manager initiates a recall operation to return the migrated file to the original file system, and when the file is recalled it returns it to the requesting application. The illustration also shows that data can be migrated between storage pools inside Tivoli Storage Manager, so in the example, it can be moved to tape pool B from disk pool A.

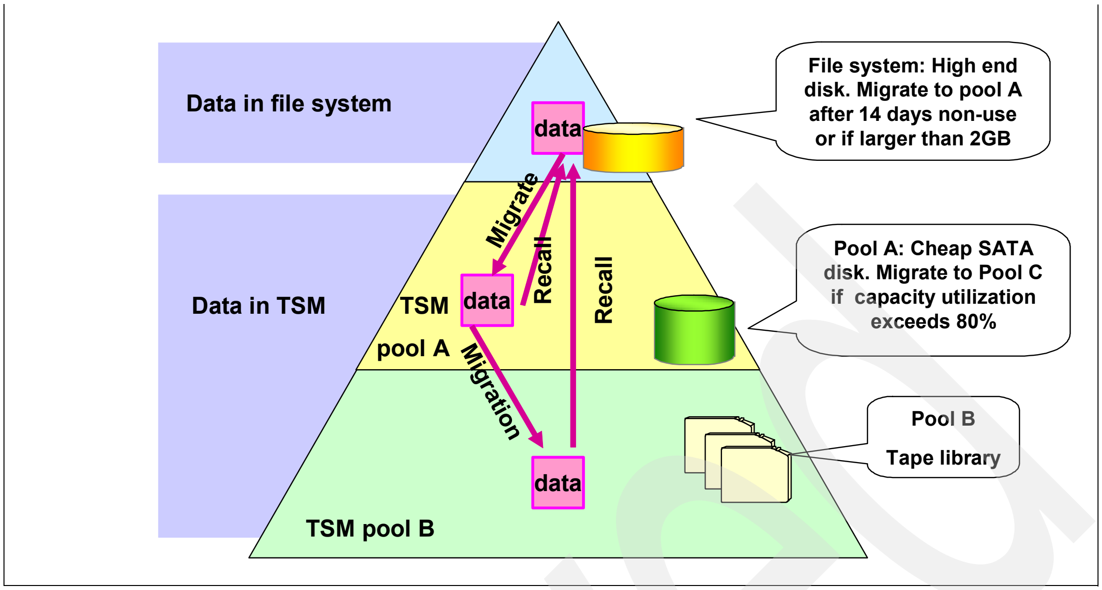


Figure 5-16 Hierarchical space management with Tivoli Storage Manager

For additional information on Tivoli Storage Manager, refer to:

http://www-306.ibm.com/software/tivoli/products/storage-mgr/

System Storage Archive Manager

IBM System Storage Archive Manager (SSAM) is special version of Tivoli Storage Manager that offers protected storage — a NERE version of Tivoli Storage Manager. SSAM is designed to make it extremely difficult to destroy data before its scheduled expiration. Short of physical destruction of the storage media or server, or deliberate corruption of data or deletion of the Tivoli Storage Manager database, SSAM helps ensure that the data it is managing remains available until its scheduled expiration date. Objects that have been archived will not be deleted from the Tivoli Storage Manager server until the retention policies set for that object have been satisfied. SSAM actively inhibits deletion of unexpired objects. With SSAM, you can suspend, and later release, this expiration date to help comply with your business and regulatory requirements. SSAM helps meet data retention and disposition regulations and policies because:

- SSAM protects data by disallowing explicit data deletion, prior to the retention criteria.
- SSAM manages data by leveraging retention policies and expiration processes.
- SSAM offers choices about where to store data by exploiting Tivoli Storage Manager's extensive device support.
- SSAM works with the Tivoli Storage Manager archive client, content management, and archive applications to make data easily retrievable.

SSAM requires a separate Tivoli Storage Manager server instance that has the data retention option turned on during server setup. Note that multiple server instances can run in the same machine. SSAM accepts data via the following client interfaces:

- Tivoli Storage Manager client application programming interface (API)
- The Tivoli Storage Manager backup/archive client

Content management and archive applications usually send data as an archive object to the Tivoli Storage Manager server via the Tivoli Storage Manager client API. No other data, such as backups, HSM data, or data base backups, can be stored on the SSAM server instance.

SSAM supports two different kinds of archive retention:

- Chronological archive retention
- Event-based archive retention

Objects retained with *chronological retention* are assigned a specific period of time to be retained, for example keep for three years. When this time period expires the object is deleted from Tivoli Storage Manager.

With *event-based retention*, expiration is based on an external event such as closing a brokerage account. SSAM supports event-based retention policy to allow data retention to be based on an event other than the storage of the data.

SSAM also offers *deletion hold and release* in order to ensure that records are not deleted when a regulatory retention period has lapsed but other legal requirements mandate that the records continue to be maintained. The SSAM deletion hold feature will prevent stored data from being deleted until the hold is released.

For additional information on SSAM, refer to:

http://www-306.ibm.com/software/tivoli/products/storage-mgr-data-reten/index.html

DR550

IBM System Storage DR550 is an integrated offering for clients that have to retain and preserve electronic business records. It is designed to help store, retrieve, manage, and retain regulated and non-regulated data. In other words, it is not just an offering for compliance data, but can also be an archiving solution for other types of data. The DR550 systems are designed as preconfigured offerings with servers, storage, and software integrated. The offerings help to preserve and retain electronic business records, either to comply with government and industry regulations, or simply because there is a business requirement for retaining data.

The DR550 is built around standard IBM components such as System p™ servers with AIX, DS4000 disk storage and IBM SAN switches. Two models are offered: The DR550 Model DR1 ranges from 0.88 to 36.88 TB raw disk storage capacity, with a choice of ethernet connections and comes in a 25U rack with lockable doors and is tape ready. The DR550 Model DR2 ranges from 6 to 168 TB of raw disk storage capacity, it offers single or dual redundant server options, ethernet connections and a synchronous or asynchronous replication option. It is delivered in a 36U rack with lockable doors and is tape ready.

Data is written to the DR550 by the content-management applications using the IBM System Storage Archive Manager Application Program Interface (SSAM API) or applications that use the Tivoli Storage Manager client directly.

File System Gateway

An optional component called the IBM System Storage DR550 File System Gateway enables many additional content-management applications to access the DR550 by adding NFS and CIFS network file access. The File System Gateway takes files that are sent using network file protocols and associates these files with an IBM System Storage Archive Manager management policy. This is done by using customer-configurable path and file naming pattern matching rules. The File System Gateway sends these files with their associated policies to the DR550 using the SSAM application program interface. Files are retrieved using the same name and directory as they were stored under.

Data retention

The DR550 supports both chronological and event-based data retention, it uses policies to manage information. Event-based retention enables management of data that has no explicit retention period (such as insurance policies) and protects records from deletion until a specific event occurs.

The deletion hold function allows a designated retention date to be suspended through a deletion-hold management feature in the event that a record or set of records must be retained for legal, audit or other reasons. The SSAM permits users to automatically archive files from their workstations or file servers to data-retention protected storage, and to easily retrieve archived copies of files to their local workstations or file servers.

The DR550 maintains data in non-erasable and non-rewritable formats. DR550 offers non-rewritable, non-erasable storage controls to prevent deletion or alteration of data stored on the system to help protect business critical information for regulatory or non-regulatory reasons. The DS4000 Enhanced Remote Mirroring option supports business continuity in the event of a disaster. The SSAM software in the DR550 provides integrated backup client that allows for backing up entire DR550 content (full or incremental) to an attached tape system.

Data migration

The DR550 supports integrated data migration because data outlives media, it enables the migration of data from older to newer media technology. DR550 is designed to support data migration capabilities to move data from disk to tape and from generation to generation while maintaining data immutability. The DR550 enables data management on multiple tiers of storage (disk, tape and optical) to provide a more cost-effective solution. It is costly to keep all archived data in tier 1 storage throughout its retention period. DR550 is designed to enable you to take advantage of a storage mix as their archiving repository to contain costs while leveraging storage technology for optimum usage.

Data encryption

The DR550 provides data encryption options (128 bit AES or 56 bit DES technology) to help companies protect their data when transmitted over the network, or saved to disk or tape. Data is encrypted prior to transmission and remains encrypted in the DR550, including backup copies. Key management can be provided by the DR550 or externally by the application. DR550 also supports tape encryption with the IBM System Storage TS1120 and LTO4 tape drives. The DR550 offers data Shredding to help protect sensitive data from being re-created after deletion.

Data backup and recovery

The DR550 supports three options for data backup and replication, as illustrated in Figure 5-17. The DR550 can be configured to back up data to tape, and you can then move the tapes offsite to a remote location for disaster recovery purposes. If the content manager archiving application supports dual writes then data can be written to two separate DR550 systems in separate locations over an IP network because the archiving application connects to the DR550 over IP. DR550 also supports DS4000 Enhanced Remote Mirror functionality for disk storage mirroring between two separate DR550 systems.

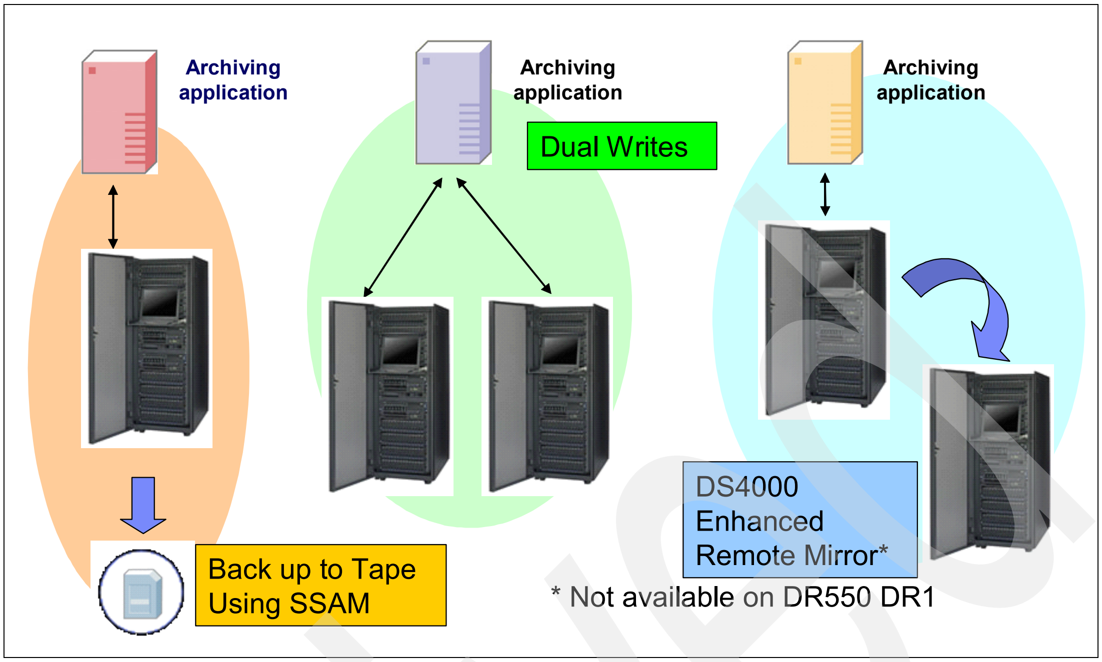


Figure 5-17 DR550 data backup and replication options

For additional information on DR550, refer to:

http://www-03.ibm.com/systems/storage/disk/dr/index.html

5.2.6 File systems

File systems are an integral part of how applications access storage systems. On UNIX operating systems everything is a file. Data gets stored by applications on underlying file systems and the data is named and stored according to some application-defined or user-defined scheme. File systems are a very useful control point in an information infrastructure because a system management application working at the file system layer, and accessing file system metadata, will know which application stored the data based on the file user and security attributes and when the data was stored and last referenced based on the file reference dates. System management applications can act on this information to perform intelligent management operations on the file such as moving it to a different storage location or storage tier to control cost or replicating it for availability.

In this section we discuss the IBM Generalized Parallel File System (GPFS). For completeness we include Grid Access Manager — software that includes file system interfaces and two storage solutions that have file system interfaces.

IBM General Parallel File System (GPFS)

The IBM General Parallel File System (GPFS) is a high-performance shared-disk file management solution that provides fast, reliable access to a common set of file data from two computers to hundreds of systems.

GPFS provides unmatched performance and reliability with scalable access to critical file data. GPFS distinguishes itself from other cluster file systems by providing concurrent high-speed file access to applications executing on multiple nodes of an AIX cluster, a Linux cluster, or a heterogeneous cluster of AIX and Linux nodes. GPFS provides storage management, information life cycle tools, centralized administration and allows for shared access to file systems from remote GPFS clusters. GPFS is capable of managing up to petabytes of data and billions of files.

GPFS scales from a two node to 2,000 nodes or more in the case of applications like modeling weather patterns. Up to 512 Linux nodes or 128 AIX nodes with access to one or more file systems are supported as a general statement and larger configurations exist by special arrangements with IBM. GPFS was designed from the beginning to support high performance computing and has been proven very effective for a variety of applications. It is installed in clusters supporting relational databases, digital media and large scalable file serving environments.

In a GPFS cluster applications access files through standard UNIX file system interfaces or through enhanced interfaces available for parallel programs.

GPFS file system administration is similar to standard AIX and Linux administration while providing extensions for the clustering aspects of GPFS. Administration functions support cluster management, quotas, snapshots, and extended access control lists. GPFS offers an SNMP interface for monitoring by external applications and supports the Data Management API (DMAPI) interface that allows storage management applications such as Tivoli Storage Manager to provide HSM support for GPFS. Tivoli Storage Manager HSM support allows a GPFS filesystem to exploit tape storage as a repository. GPFS file systems can be exported to clients outside the cluster through NFS or Samba. It also offers clustered NFS tools include monitoring of file services, load balancing and IP address fail over.

GPFS is fault tolerant and can be configured for continued access to data even if cluster nodes or storage systems fail. This is accomplished though clustering features and data replication. Replication allows for continuous operation even if a path to a disk or a disk itself fails.

GPFS provides for Information Lifecycle Management (ILM) with the introduction of storage pools, policy-based file management, and filesets. A file in a GPFS filesystem maintains its path and name regardless of where it is placed by GPFS policy based data management, therefore the application does not have to track file name changes. GPFS introduces the following storage management constructs:

- Storage pools
- Policies
- Filesets

GPFS *storage pools* allow you to manage your file system’s storage in groups. You can partition your storage based on such factors as performance, locality, and reliability. A storage pool is a collection of disks with similar properties that are managed together as a group, for example, you can have a high-performance disk pool and a low-cost disk pool. Files are assigned to a storage pool based on defined policies. Files can be moved between storage pools by changing the file’s storage pool assignment with commands. The file name is maintained unchanged.

GPFS *filesets* provide a means of partitioning the namespace of a file system, allowing administrative operations at a finer granularity than the entire file system. A fileset is a subtree of a file system namespace that in many respects behaves as an independent file system. Filesets provide a means of partitioning the file system to allow administrative operations at a finer granularity than the entire file system:

- You can define per-fileset quotas on data blocks and inodes. These are analogous to peruser and per group quotas.
- Filesets can be specified in the policy rules used for placement and migration of file data.

Policies and rules

GPFS *policies and rules* provide a means to automate the management of files using predefined policies and rules. GPFS supports the following policies:

- File placement policies are used to automatically place newly created files in a specific storage pool.
- File management policies are used to manage files (migrate or delete) during their lifecycle by moving them to another storage pool or deleting them.

A policy rule is an SQL-like statement that tells GPFS what to do with the data for a file in a specific storage pool if the file meets specific criteria. A rule can apply to any file being created or only to files being created within a specific fileset or group of filesets.

GPFS is a powerful enabler to build a highly available tiered storage environment with both diverse type of disk and tape tiers through Tivoli Storage Manager HSM. An ECM application can store data on the filesystem and then let the filesystem manage the data based on predefined rules and policies. The ECM application can also interact directly with GPFS through commands to request a file to be moved to a different tier based on events that occur inside the ECM system.

For additional information on GPFS, refer to:

http://www-03.ibm.com/systems/clusters/software/gpfs/index.html

Scale out file services

IBM also offers a pre-integrated and pre-tested file serving solution based on GPFS called scale out file services (SOFS). SOFS are designed to eliminate your information sharing challenges through swift implementation of a highly scalable, global, clustered NAS system.

For additional information on SOFS, refer to:

http://www-935.ibm.com/services/us/its/html/sofs-landing.html

Grid Access Manager (GAM)

IBM System Storage Multilevel Grid Access Manager Software (GAM) offers an open high performance grid architecture designed to deliver data protection, information lifecycle management, simplified storage management and multi-site data access. These features offer:

- Simplified management and improved storage utilization, with excellent performance
- Data protection and improved business continuity
- Support for global access, multi-site operation

GAM Software enables customers with single or multiple sites and with fixed content and reference data storage requirements to improve storage utilization and investment across sites via an enterprise-wide, fault-tolerant storage grid with real time failover capabilities. GAM can help protect enterprise data via the use of automated replication, lifecycle management and digital signature verification functionality.

GAM delivers an automated and virtualized storage and data management layer that can be deployed locally or across multiple facilities and even on heterogeneous storage hardware. Grid Access Manager Software achieves this by presenting itself to enterprise applications as a single storage file system that can aggregate underlying storage silos into a single enterprise storage pool—even if these silos are spread across multiple facilities and consists of heterogeneous storage media. This highly automated system can allow IT administrators to focus on storage planning and infrastructure improvements and eliminate many manual administration tasks.

Enabling features

GAM offers simplified management and improves storage utilization. It operates across locations, multiple storage tiers and storage devices, GAM supports tiered storage architectures. Key enabling features include:

- Intelligent Information Lifecycle Management, GAM enables the creation of information lifecycle management (ILM) policies that govern the geographic location and placement of data in a multi-tier, multi-facility storage system. Intelligent ILM can help deploy the right data in the right place, on the right media, at the right time based on its relevance and value to the organization.
- Scaling because capacity, performance and geographic scope of GAM can be increased by adding new nodes. As nodes are added typically in a non-disruptive manner, GAM can continue to present itself as one large scalable computing system. GAM automatically re-configures the environment to take advantage of the new resources.
- GAM supports automated data migration, it can automate the movement of data from obsolete resources to new resources—reducing or eliminating manual, disruptive data migration operations.
- GAM supports industry standard access protocols such as CIFS, NFS and HTTP.
- GAM offers proactive monitoring and management by continuously monitoring the health of the system including storage, servers and network and pro-actively alerts administrators if issues arise. Administrators can monitor and manage the grid from across the network using the single pane, Web-based management console. More than 200 real-time and historical reports, allow administrators to track CPU, network, storage resources and digital data assets themselves.

GAM supports data availability and business continuity by creating multiple copies of data and automatically replicating them geographically or to different storage tiers based on customer rules so they are no longer vulnerable to potential data loss issues. Designed to support continuous operations even in the event of failure, GAM provides automated recovery to help maintain business resiliency. The solution also supports continuous operation during hardware refreshes, capacity upgrades and data migration. GAM provides a fault tolerant, resilient and self-healing solution. Storage services can remain available even after hardware, storage and network failures occur or where entire sites are completely destroyed. This is illustrated in Figure 5-18.

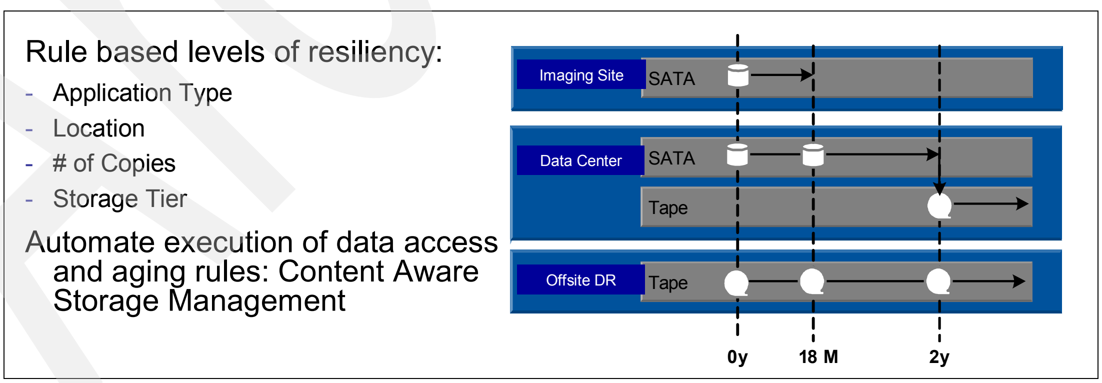


Figure 5-18 GAM creating multiple replicated copies of data

GAM protects the integrity and authenticity of stored data using digital fingerprints. Data integrity is pro-actively monitored and verified when data is stored, replicated, restored and retrieved. Integrity checks can be performed on demand and if errors are encountered, Grid Access Manager Software can automatically create another replica from a known good copy.

GAM offers multi-site operations, key enabling features are:

- GAM provides n object store with global name space, an object based storage system that can scale to petabytes (capacity) and billions of objects (object count) in a single unified system that can span multiple sites and tiers. A global name space gives access to stored objects using globally unique object identifiers, which can be accessed from any grid resource — regardless of location or storage tier. This eliminates storage silos to enable efficient and effective multi-site access.
- WAN Optimized Replication — GAM provides policy based N-way data replication among sites and heterogeneous storage hardware. The replication semantics are highly optimized for WAN operation and designed to recover from congestion or link failure.

IBM System Storage Multilevel Grid Access Manager Software

IBM also offers an integrated hardware and software solution based around GAM software called Grid Access Manager Software (GMAS). The GMAS solution supports both disk and tape storage tiers and also offers connection to the IBM DR550 protected storage solution.

GAM is a very good solution when you require a mechanism to deploy data to multiple different locations and move it between diverse storage tiers.

For additional information, refer to:

http://www-03.ibm.com/systems/storage/software/gam/index.html
http://www-935.ibm.com/services/us/index.wss/offering/its/a1027917

5.2.7 TotalStorage Productivity Center for storage management

The IBM TotalStorage Productivity Center is a suite of storage infrastructure management tools that assists in reducing the complexity of managing your storage environments by centralizing, simplifying and automating storage tasks associated with storage systems, storage networks, replication services, and capacity management.

TotalStorage Productivity Center is a single integrated solution that combines the assets, capacity, performance and operational management, traditionally offered by separate Storage Resource Management, SAN Management and Device Management applications into a single platform homogenous platform.

TotalStorage Productivity Center provides of four separate components:

- TotalStorage Productivity Center for Data
- TotalStorage Productivity Center for Fabric
- TotalStorage Productivity Center for Disk
- TotalStorage Productivity Center for Replication

TotalStorage Productivity Center for Data

TotalStorage Productivity Center for Data helps monitor, manage and automate capacity utilization of your file systems and databases. It includes enterprise reporting, policy based management, automated file system extension, Tivoli Storage Manager integration, database capacity reporting, and chargeback capabilities.

These are the main functions of TPC for Data:

- Discover and monitor disks, partitions, shared directories, and servers.
- Monitor and report on capacity and utilization across platforms to help you to identify trends, based on historical data, and prevent problems.
- Monitor storage assets associated with enterprise-wide databases and issues notifications of potential problems.
- Provide a wide variety of standardized reports about file systems, databases (using the Data Manager for Databases function), and storage infrastructure to track usage and availability.
- Provide file analysis across platforms to help you to identify and reclaim space used by non-essential files.
- Provide policy-based management and automated capacity provisioning for file systems when user-defined thresholds are reached.
- Generate invoices that charge back for storage usage on a departmental, group, or user level (using the Data Manager for Chargeback function).

TotalStorage Productivity Center for Fabric

TotalStorage Productivity Center for Fabric monitors and manages the SAN fabric. It helps the administrator simplify the task of SAN management and configuration and maintain optimal SAN availability and performance. TPC for fabric provides these features:

- Topology views offer both host-centric and device-centric views that can be displayed graphically or in a hierarchical format.
- The centralized topology view is based on information from a number of sources, such as Element Management tools, device logs, and SNMP traps that are continually monitored correlated to determine the current SAN configuration and topology
- TotalStorage Productivity Center for Fabric provides basic diagnostic capabilities to show which resources are impacted by an availability or performance issue in the SAN.
- SAN configuration and change via SAN zone control
- Inband and outband SAN discovery and control.

TotalStorage Productivity Center for Disk

IBM TotalStorage Productivity Center for Disk is designed to help reduce the complexity of managing SAN storage devices by allowing administrators to configure, manage, and performance monitor storage from a single console. TPC for Disk offers:

- Configuration of multiple storage devices from a single console
- Monitor and track the performance of SAN attached Storage Management Interface Specification (SMI-S) compliant storage devices
- Enable proactive performance management by setting performance thresholds based on performance metrics and the generation of alerts
- Committed to open standards This architecture of this solution is based on the Storage Networking Industry Association (SNIA) SMI-S and supports compliant SAN components
- Manage heterogeneous storage systems from a single console
- Discover and configure storage systems
- Monitor and manage performance, configure performance thresholds and generate alerts
- Store capacity and performance statistics and generate point-in time and historical reports
- Perform volume contention analysis

TotalStorage Productivity Center for Replication

TotalStorage Productivity Center for Replication provide management of FlashCopy, Metro Mirror and Global Mirror capabilities for the IBM ESS Model 800, IBM DS6000, and IBM DS8000. It also manages FlashCopy and MetroMirror for the SAN Volume Controller. TotalStorage Productivity Center for Replication simplifies management of advanced copy services by:

- Automating administration and configuration of these services with wizard-based session and copy set definitions.
- Providing simple operational control of copy services tasks, including starting, suspending and resuming.
- Offering tools for monitoring and managing copy sessions.

TotalStorage Productivity Center functional packages

TotalStorage Productivity Center is orderable in different functional packages.

The IBM TotalStorage Productivity Center Basic Edition is included with various IBM System Storage disk products, including the DS8000, DS6000 and DS4000. It also is available with the SAN Volume Controller and select IBM tape libraries. It is designed to:

- Discover and configure IBM and heterogeneous SMI-S supported devices
- Perform event gathering, error logging and launch device element managers
- Provide basic asset and capacity reporting
- Display an end-to-end topology view of your storage infrastructure and health console
- Enable a simple upgrade path to TotalStorage Productivity Center Standard Edition

TotalStorage Productivity Center Standard Edition provides all the management capabilities to better manage your heterogeneous storage infrastructure from application to back-end storage system disk at a single bundled price. It consists of:

- TotalStorage Productivity Center for Data
- TotalStorage Productivity Center for Fabric
- TotalStorage Productivity Center for Disk

TotalStorage Productivity Center for Replication is packaged and priced separately from TotalStorage Productivity Center Standard Edition. In addition, TotalStorage Productivity Center for Replication is available in multiple complementary packages for the open environment and System z:

- TotalStorage Productivity Center for Replication Two Site BC
- TotalStorage Productivity Center for Replication Three Site BC
- TotalStorage Productivity Center for Replication for System z

The TotalStorage Productivity Center suite of products offers a comprehensive set of capabilities for the information availability and information retention themes. It offers simplified and more effective management of the storage infrastructure and can be of use when managing offline storage as TotalStorage Productivity Center for Data can interface products such as Tivoli Storage Manager to automatically archive old files.

For more information on TotalStorage Productivity Center, refer to the following site:

http://www-03.ibm.com/systems/storage/software/center/replication/index.html

System Storage Productivity Center (SSPC)

The IBM System Storage Productivity Center (SSPC) is an integrated and pre-configured hardware and software solution that helps provide extended device configuration capabilities for heterogeneous devices while also consolidating management to a centralized platform. The SSPC software is designed to allow you to manage a storage system in the context of the broader storage environment. The SSPC now integrates enterprise SAN management with the low-level device configuration functions that were previously only available in standalone consoles for DS8000 and SVC. With this integration, DS8000 and SVC administrators now have access to an expanded toolset.

The IBM System Storage Productivity Center (SSPC) includes the following pre-installed (separately purchased) software:

- IBM TotalStorage Productivity Center (separately purchased)
- IBM System Storage SAN Volume Controller (CIM Agent and GUI)

For additional information on the SSPC, refer to:

http://www-03.ibm.com/systems/storage/software/sspc/index.html

6

IBM FileNet P8 Platform

The IBM content management heritage spans over 20 years. This experience has resulted in the creation of IBM FileNet P8 Platform, the core technology behind IBM FileNet ECM.

In this chapter, we discuss the IBM FileNet P8 Platform. As a standards based, open ECM solution, P8 allows broad flexibility in technology and implementation, the interoperability of which enables limitless integration opportunities between disparate business systems. The unrestricted design of P8 allows technology managers to retain their choice of infrastructure without relinquishing their freedom to change vendors in the future.

6.1 Overview

The IBM P8 Platform, as part of an Information Infrastructure, is deployable as a stand alone solution or as a robust framework to support highly specialized applications. Using IBM FileNet authoring tools, business analysts are empowered to shape and mold business processes using a drag and drop interface. This allows analysts to automate and track activity between business entities without requiring a deep understanding of the implementation. These processes, at a click of a button, are exposable as a Web services capable of delivering cradle to grave SOA solutions for any P8 powered business service.

IBM P8's capabilities are considerable out of the box, yet it is designed to be tailored to specific organizational requirements. P8 ships with tools APIs to build robust and resilient applications that unify data between multiple business systems. This broad flexibility allows business leaders to concentrate on strategic direction rather than wrestling with complex integration issues. A strong toolset further allows administrators to brand the P8 user interface and a Model View Controller design pattern allows developers the flexibility to integrate disparate business systems and unify data and processes. From insurance claims processing to refinery asset management, P8 provides a resilient, scalable platform to create industry specific solutions tailored to specific business requirements.

By design, nearly every tier in an IBM P8 ECM solution offers some degree of customization to ensure the capabilities of the ECM solution are in line with the requirements of the organization. The consequence of this capability affects how the storage infrastructure is used, and the volume of storage required to sustain the solution. Business system development is a continually evolving process and, in turn, the storage architecture must possess the capability to accommodate change as business requirements mandate.

6.2 IBM P8 Platform core applications

IBM FileNet P8 Platform is a suite of applications for delivering a base line ECM solution. Three core applications comprise the framework for the various add-on products IBM FileNet supports and is the foundation on which all IBM FileNet ECM applications are built. Although Content Engine is the primary component requiring storage, Application Engine and Process Engine strongly influence how the content is stored and retrieved.

6.2.1 Application Engine

The Application Engine architecture diagram shown in Figure 6-1 consists of two components, a J2EE™ Web Application and a standalone communication stack used to send and receive messages between other FileNet components. The Application Engine Web interface ships in two flavors, Workplace™ and Workplace XT (not shown).

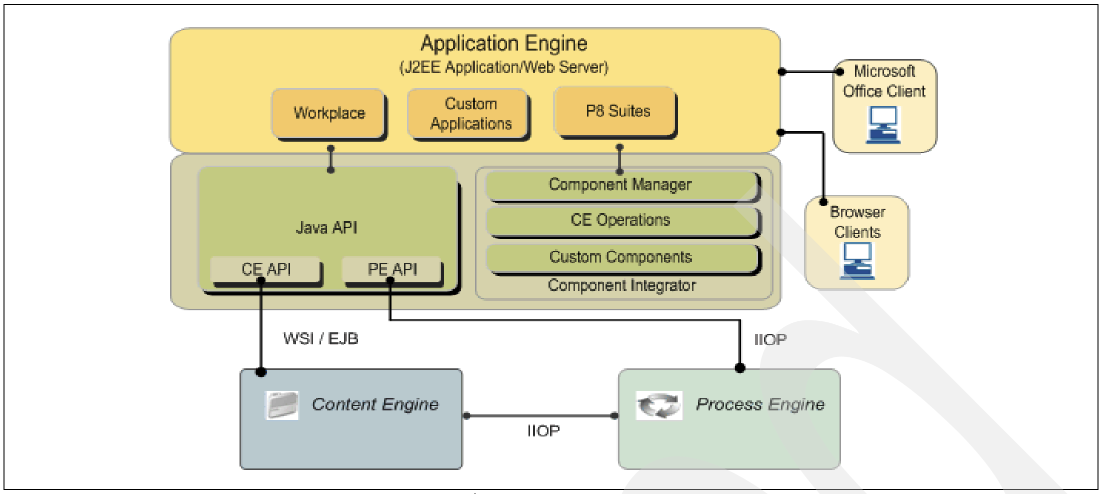


Figure 6-1 Application Engine Architecture[1]

Workplace is a Web interface written using a Model View Controller design pattern and is intended for customers wishing to modify and tailor the look and feel of their ECM solution. Workplace XT, conversely, is an AJAX driven Web interface optimized for out of the box, content and workflow management with a focus on maximizing operational efficiency by minimizing the number of clicks and page loads required to store, retrieve, and act on content.

Both Workplace and Workplace XT communicate using the same Content and Process API stack available to third party development. This ensures all of the capabilities observed in the user interface are available for use by developers. The Component Integrator shown in Figure 6-1 represents not the Component Integrator itself but the administrative framework used for managing Process Engine.

How a user interacts with the user interface influences the use of storage. Those familiar with high volume Web environments will have an immediate appreciation for how the user interface influences the consumption of resources. For example, Web sites predominately generate revenue from advertisers who place ads amongst their content. The amount the advertiser pays is usually based on how many people view the ad and many people subsequently click though to the advertisers site. Subtle changes in ad placement and interface design influence how the viewer will react to the ad. The same logic applies to the Application Engine and storage. If the user interface allows content to be uploaded with minimal user interaction, the storage consumption rate will be greater than a user interface requiring interacting with multiple fields and pages such as would be found in a wizard. In short the fewer interactions required to upload content, the more likely a user will take advantage of its capabilities and upload more frequently.

Use of storage is described in terms of a use pattern and a consumption rate. These factors affect the strategy required to successfully manage the storage platform and must be analyzed to ensure the storage architecture is capable of managing the use pattern.

6.2.2 Content Management: Content Engine

Figure 6-2 indicates that Content Engine, like Application Engine, runs in an application server, not as a Web application, but as an Enterprise Java Bean. Content Engine is responsible for storing and retrieving content, its metadata and obfuscating storage.

[1] IBM. IBM FileNet P8 System Overview

Collectively, a piece of content and its associated metadata are known as an object. Objects are stored in a storage obfuscated repository, termed aptly, an object store.

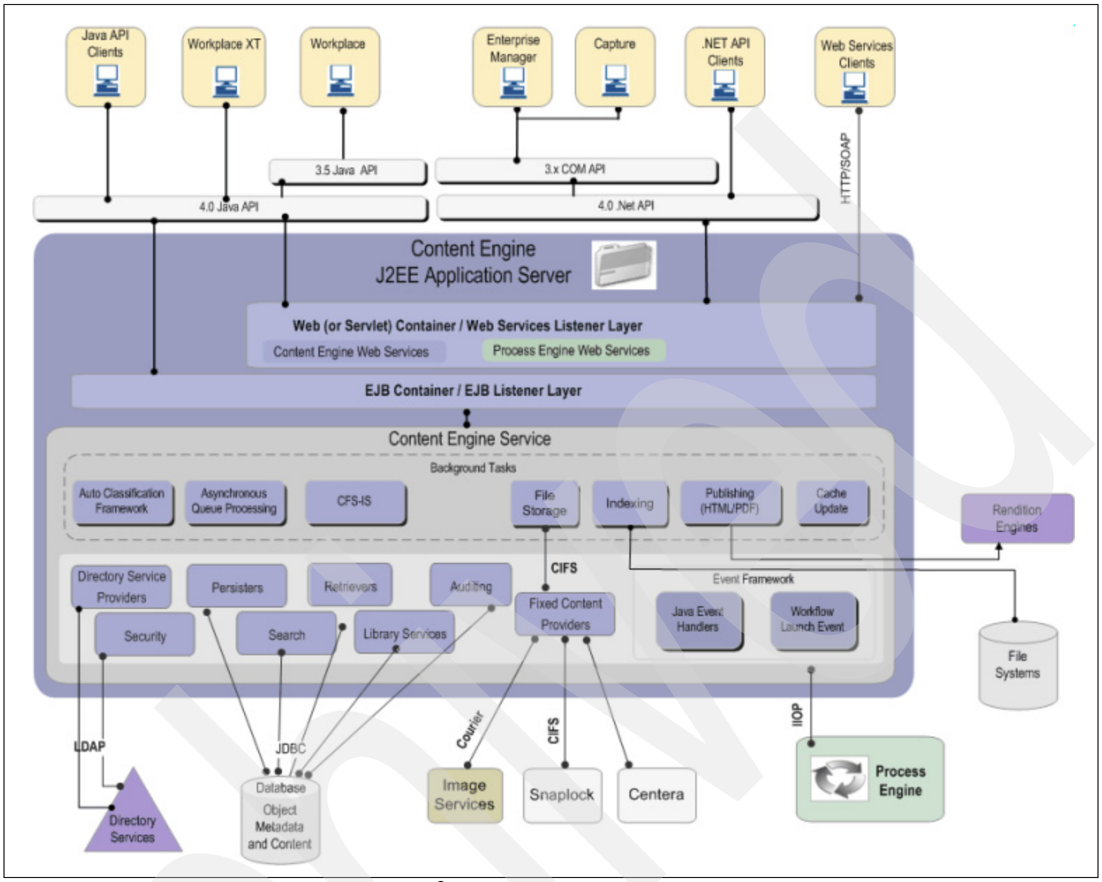


Figure 6-2 Content Engine Architecture[2]

Although every effort has been made to shield users from storage complexities, administrators have been given considerable freedom to construct a solution to best meet organizational requirements.

An object store is logical black box storage medium both to the user and the developer. Neither is aware of how the content is managed, on which systems the data resides, or how it is made available. Content Engine is an information service that allows storage complexities to be managed exclusive of the presentation and delivery of the data. Beneath the surface however, business and storage logic is administratively managed while maintaining transparency to the user.

Content Engine provides the capability of controlling key aspects of object management though configuration without requiring extensive programming to an API. Managing objects in this manner addresses two challenges transcending all institutions of the information age, reducing time to market, and increasing business agility. By implementing storage management beneath the content API, policies can be modified without introducing application changes. The agility provided by this design empowers administrators to intelligently manage object storage as it is relevant from a business perspective, freeing developers to focus on the development of the business solution. The storage implementation of Content Engine allows administrators and developers to work more independently and deliver solutions faster.

[2] IBM. IBM FileNet P8 System Overview

For example, content created by an accounts receivable group might be handled differently from content received from a human resources group. An administrator can configure the storage location for each independently without introducing an application change. A developer does not have to account for this change because it is independent of the APIs used to develop the solution. Thus, the business requirements are addressed with fewer individuals and sooner than would be required if more people were involved.

Content Engine's freedom of implementation requires considerable forethought in design and architecture. Understanding the details of how data is stored and retrieved is predicate to understanding how the storage system plays a vital role in a successful Information Infrastructure. When an object is requested from an outside entity the content API calls the Content Engine EJB™ retrieve the object. The Content Engine EJB is also used to store and retrieve Process Engine objects for workflow related data. The idiom of an object is heavily intertwined in all aspects of P8 application components and is the foundation of the platform architecture.

Object store conceptual model

From a administrator's view point an object store is comprised of a database instance and one or more storage areas. By default, content is stored along with its metadata in the database. Databases being sub-optimal for storing files, are usually not where content is typically stored for production use. Content Engine instead offers the ability to store content using a file system or a fixed content device. Locations that store data are known as Storage Areas and a logical collection of storage areas are known as a Storage Policy. Determining how the content is stored requires a brief discussion of what a Content Engine Object is and how it relates to content management.

An object store functions in many respects like a filing cabinet. Documents are placed in folders to organize the contents. A single filing cabinet is rarely enough to store the all of the documents that require the use of multiple filing cabinets. As the number of filing cabinets grows, so does the difficulty in finding a specific document. A document of interest could be in one of many filing cabinets in one of many folders. A cue card system, indexing the types of documents and their titles, eases the burden of having to open each and every cabinet to find a document. In Content Engine terms a filing cabinet is a storage area, multiple filing cabinets are a storage policy, and the index is the object store database. We can extend the filing cabinet analogy further by having multiple rooms of filled with filing cabinets, which in Content Engine terms would be synonymous with having multiple object stores.

Content saved in an object store is an object regardless of the nature of the content. Object is a generic term used to describe the logical representation a piece of metadata (or a collection of properties) and the content itself. Unlike a filing cabinet, an object store is a structure containing content and metadata, where each object is an instance of a class. This concept is borrowed from the notion of an object as it relates to object oriented programming. A class defines a fixed set of properties and actions. An object is an instance of a class whose property values are unique to the object.

The base class for content is the Document Class. All documents in the object store inherit the properties of the class to which they belong, by default the root Document Class is configured to store content in the database. Content can also be stored in a Fixed Content Device or in a file system. The class based storage capability allows content to be directed to the storage system that is most appropriate to archiving the data as it relates to its business purpose. This capability is completely transparent to users. As data is uploaded, Content Engine silently routes the documents based on their class to the appropriate storage medium.

The user is exposed only to a single unified namespace irrespective of where objects are stored and does not have to be aware of the underlying business logic. This capability is implemented well beneath the presentation tier in the Content integration layer and shields the storage logic from the developer and the user. As business requirements change, the routing logic can be amended to reflect changes without having to retool user interface enhancements or integration code.

Often times objects are not loaded into Content Engine with Application Engine but are instead loaded by add-on applications and APIs that complement how users store data before the introduction of the ECM application. The following is a list of P8 add-on application which contribute to the storage foot print of an IBM P8 system.

- IBM FileNet e-mail Manager
- IBM FileNet Records Crawler
- IBM FileNet Capture
- IBM Information Integrator: Content Edition
- Programatically:
  - Content Engine Java Client API
  - Content Engine Net Client API
  - Content Engine COM Client API
  - Content Engine Web Services

6.2.3 Business Process Management: Process Engine

Business Process Management provides services for managing all aspects of business processes (also called workflows), such as process execution, process routing, rules management, process simulation and modeling, and workflow analysis. Process Engine, the core software that provides these services, allows creation, modification, and management of workflow implemented by applications, enterprise users, or external users (such as partners and customers).[3]

As shown in Figure 6-3, Process Engine unlike it counterparts, does not run in an application server. Instead it runs as a set of daemons written in C and a Java communications stack. Process Engine is responsible for all workflow related tasks and utilizes an external database different from Content Engine to track workflow data. Workflows created using Application Engine are transferred to the Process Engine where they can be launched. The workflows themselves are stored in the Content Engine where presentation layer components are able to launch them.

[3] IBM. IBM FileNet P8 System Overview

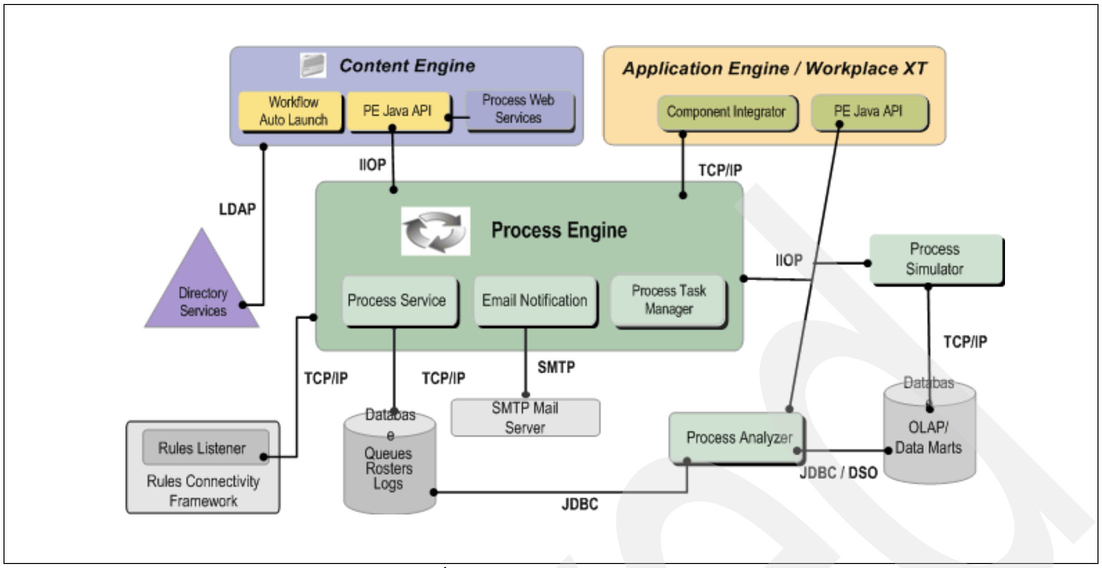


Figure 6-3 Process Engine Architecture[4]

A workflow, on its own, does not impact the storage foot print of an IBM P8 environment to the same degree as Content Engine. It does however influence it by broadening the audience of content in an automated way. For example, an accounting reconciliation process might involve a versionable spread sheet that is routed to different business units so that each can add their data to the document one after another. A versionable document is one where the object content is copied each time it is checked in. This provides an audit trail and the capability to revert to an older version of a document. The purpose of versioning in this example is to maintain an audit trail so that business units cannot change the spreadsheet data for other business units that have contributed to the document. Each time a document is versioned (checked out and checked in), a copy of the document is stored in the object store. If a 512 kilobyte document is versioned ten times, the storage the object consumes is about five megabytes. While our accounting example might be overly simplified, it does demonstrate how a single workflow can grow the storage footprint of an object store.

If the capabilities of Process Engine are leveraged heavily using document versioning as previously described, an otherwise singular document object will consume considerably more disk than it would were it were stored without versioning. Great care must be taken when implementing workflow to ensure the storage platform is sized to accommodate a potentially accelerated growth rate and that volumes are easily expandable to accommodate changes affecting the growth pattern. Assume for a moment that the previously mentioned accounting reconciliation workflow did not implement versioning capability. The original storage consumption rate would be calculated and storage volumes created to accommodate a shallow growth rate. If at a later date a business change requires having to version this data, the previous growth pattern would not longer apply and the storage would have to be changed to reflect the new direction in policy.

Great care must be taken to ensure the capabilities of the infrastructure and its implementation do not artificially restrict the capabilities of the business. This is achieved by ensuring the storage supporting the application can keep pace with the changes in the ECM solution. The accounting example described a process whose growth rate increased ten fold. A re-engineering of the application and infrastructure at this juncture would be highly disruptive and prevent the business from utilizing the software the way it is required. In short design oversights can force an organization to work around the infrastructure instead of

4 IBM. IBM FileNet P8 System Overview

having the infrastructure work around them. In many respects an inappropriate choice of storage could prevent the business from acting on critical changes and potentially cause an interruption preventing the organization from acting on market opportunities.

6.2.4 Content consumption drivers

IBM FileNet P8 platform integrates with a number of add-on products, which cause storage to be consumed at accelerated rates and different in use patterns. Use of these add-on products will influence consumption rates, read/write ratios, and content life span. As such, each must be independently considered, accounting for the business problem to which they are applied, and the changes that they cause all factor into the choice and configuration of the storage platform. Each of the following add-on applications have considerable storage implications:

- IBM FileNet E-mail Manager
- IBM FileNet Records Crawler
- IBM FileNet Capture
- IBM FileNet Integration for Microsoft Office
- Content and Process APIs and Web Services

IBM FileNet E-mail Manager

Using E-mail Manager, an organization can manage e-mail content as a part of a comprehensive Enterprise Content Management infrastructure. E-mail Manager, used in conjunction with an IBM FileNet repository, enables organizations to capture, organize, monitor, retrieve, and share e-mail content when and where it is required, including using captured e-mail content to initiate, and participate in, automated business processes. Using E-mail Manager, records are searchable and can be purged as with any other corporate record. E-mail Manager simplifies and automates the process of capturing e-mail messages, archives them as business records, and provides proof of compliance with industry and government regulations[5].

E-mail Manager's capabilities help eliminate the capture of duplicate e-mail messages and ensure that copies of e-mails, with or without attachments, will be written to an object store only once. By eliminating duplicate e-mails before they are added to a repository, e-mail Manager provides better utilization of storage and simultaneously lowers overall storage costs[6].

Managing extreme volumes of email, as is typical of most organizations, requires considerable forethought with regards to storage. Archived e-mails in most cases will not be used outside of compliance requirements and as a record they have limited monetary value (that is they cannot be used to generate revenue). E-mail volumes grow quickly and when stored as records, the ability to enforce mail box limits or purge data as one would in a mail store, is lost (for example SEC rule 17a-4 requires retaining e-mail for a minimum of three years[7], much longer than a mail box would typically go without purging or enforcing quotas). Compounding the growth rate, is the content index, which allows the content of the e-mail to be quickly searched. As much as an additional 50% more disk space must to be allocated to accommodate this feature. The volume level might require a storage system that can quickly archive the data and move it to a more cost effective storage medium for long term archival, while at the same time providing acceptable performance for the index.

E-mail also bears the burden of relevance. The average user receives numerous e-mail messages that are relevant to their work, but others are personal in nature and some are completely unsolicited. While the term “unsolicited e-mail” is broadly understood to be spam,

[5] IBM. IBM FileNet P8 System Overview
[6] IBM. IBM FileNet P8 System Overview
[7] http://www.sec.gov

company memos and industry mail lists are not the target of spam filters. That said cleverly crafted e-mails, company memos and mail list activity can be unintentionally committed as records that will contribute consumption of storage. While E-mail Manager is able to apply business logic to filter some of these messages, a portion will pass though and their impact must be accounted for. These considerations assume an organization has authority to determine which e-mails are declared as records in the first place, as some will be required to declare all e-mails as records regardless of their relevance or origin.

The majority of e-mails, from a records standpoint, present no value apart from having to retain them for compliance purposes. That is until an audit or regulatory investigation requires this data to be available. The challenge is in how to store the data in a way that is cost effective yet is acceptably accessible for audit. The infrequency the e-mail record access requires storing this data on a storage system optimized for space not performance. Conversely, poor performing storage could prolong a lengthy and expensive discovery. IBM storage and ECM resolves this issue by moving data between tiers of disk relative to its access pattern. From a file system perspective the name space is unaltered and therefore completely transparent to how the data is stored. This solution allows an organization to enjoy the cost savings inexpensive storage without sacrificing performance for the short periods it is required.

IBM FileNet Records Crawler

Organizations of all sizes require effective records management in order to be compliant with external and internal regulations. Hundreds of millions of documents reside uncontrolled on network file shares where they are disorganized, unsecured, and often duplicated. With so many users and processes are contributing daily to the massive accumulation of business critical documents, enforcing corporate records policies is an increasingly challenging issue.

In light of this immense growth in file volumes, many organizations seek ways to enforce records management for documents on network file shares. The benefits of centralized records management are balanced with the flexibility of decentralized file shares, ensuring that employees or business processes can continue to use them without disruption to existing procedures. The challenge lies in bringing those documents on the network file share drives under control in a cost effective manner, while ensuring they are handled in accordance with corporate and regulatory policies[8].

Records Crawler provides the capability to continuously monitor NAS shares and effectively import large unstructured pools of content into a FileNet repository. Records Crawler supports both IBM FileNet P8, IBM Image Manager Active Edition, as well as IBM Content Manager 8. This capability allows an organization to get a jump start on migrating content from loosely structured file shares in to a highly structured ECM application, without having to start a long and costly import effort[9].

File shares, as with e-mail, have storage implications that must be considered to effectively manage and store the data. Files originating on NAS storage do not indicate to the application the business value they represent. Some contextual information, such as the folder the document was found in, or permissions of the file, can provide clues as to the file's value, but not in a way that definitively classifies the information. The nature of the data therefore defeats the class based storage policies built into Content Engine.

[8] IBM, An ECM Functional Expansion: The Benefits of Centralized Records Management with the Flexibility of Decentralized File Shares

[9] IBM, An ECM Functional Expansion: The Benefits of Centralized Records Management with the Flexibility of Decentralized File Shares

IBM storage can intelligently manage the data by quantifying its value based on a use pattern. As users manipulate the data the underlying storage is capable of moving the data to the most appropriate location. This occurs without changing the file structure as it appears to the file system or to the ECM application. This functionality allows the user activity to define the value of the data and naturally complements how the community interacts with the content while the business enjoys the cost benefit of less expensive storage.

IBM FileNet Capture

Prior to the advent of ECM, documents were either delivered electronically in the form of a word processor document or spread sheet, or arrived in a hard copy format and had to be entered into an application database by hand. IBM FileNet Capture provides the capability to enter documents in these forms automatically, with minimal human intervention.

IBM FileNet Capture provides the capability to add documents to a Content Engine or IBM Image Manager Active Edition (IMAE) repository arriving by scanning, faxing, and batch file import. By automating this labor intensive process, Capture reduces operational costs and accelerates the rate information can be consumed and therefore acted upon. Supporting a wide variety of scanners from low volume personal scanners to high volume production scanners, Capture can import data from nearly any twain device. In addition, the FaxEntry feature of the Capture product provides an automated method for capturing inbound faxes. IBM FileNet Capture provides an end to end solution for storing paper documents in an ECM solution.

Capture also eases the burden of manual paper retrieval. Without Capture paper records must be retrieved by physically locating the document before it can be acted upon. This manual people driven process dramatically retards the efficiencies of processing this data, is subject to discretion of individuals, and increases the risk of loosing documents. As such, there is no practical way to guarantee and certify paper records are handled in accordance with company policy.

When paper records are stored electronically, and managed in the ECM solution all of the capabilities of ECM are directly applicable to those records. Faxes and paper mail can be managed as auditable records eliminating the necessity to retrieve documents from a warehouse and flagging paper documents for disposal. This is especially important for paper records received from the general public.

Institutions with a rich heritage of paper based correspondence stand to benefit most from the Capture product, but must also be aware of the impact on storage. Over and above the consideration of importing their legacy paper records, these institutions also likely receive thousands of faxes and postal letters a day. These factors have implications from a storage perspective and how employees use the documents plays a considerable factor as well. The storage infrastructure must be implemented such that the rate at which paper documents arrive will not overwhelm the capacity of storage to save said documents. Furthermore allowances must be made to allow for the expansion of the storage when changes in the organization cause an influx of paperwork such as a new form or an amendment to an exiting paper form.

One such example of how paper injestion rates change is the housing boom of the mid 2000s. During this time levels of mortgage transactions were orders of magnitude larger than in previous years. A company unable to handle this influx of documents would flounder trying to re-engineer the storage to handle the overhead. A storage system capable of growing as the business grows provides assurance that the ECM solution can withstand changes in the document injestion rate as the market demands.

IBM FileNet Integration with Microsoft Office

Files from Microsoft Office applications can be checked in and out of an object store as easily as they the can be saved to the desktop. The Microsoft Office Integration option for P8 adds context items to the Office application menu bars and walks the user though saving their content to an object store. This functionality encourages adoption because the user does not have to learn a new interface. Instead they have the option of using a familiar interface to store and retrieve their documents without having to open a Web browser.

Content and Process APIs and Web Services

IBM FileNet Applications offer a rich set of tools for building ECM solutions that are tailored to the individual requirements of an organization. For some organizations, Workplace, Workplace XT and other interface tools are too general to achieve the results required. The Content Engine and Process Engine API stacks are the same ones Workplace is written against, allowing developers to leverage any feature found in the user interface components in their own applications.

The APIs available for IBM Content Engine are written in Java and expose .net interfaces using bridging code. All of the classes available to Java are bridged to ensure maximum interpretability. Languages other than these are also capable of creating custom applications by utilizing the Content Engine Web Services interface. The opens up FileNet applications to languages not having a native API to develop against such as Perl, Python, Ruby and PHP.

The storage implications for applications written to an API are nearly limitless and will likely change over time as development of a custom application matures or as business requirements change. Making any claim as to the impact to storage would be highly speculative except to say that developers, ECM, and storage architects must be aware of how their decisions affect one another to accurately plan for the long term utilization of storage.

6.3 Object storage architecture

IBM P8 Platform defines a content repository as an object store, where an object is a general term describing content belonging to a class or group. An object store architecturally consists of a database instance storing object metadata and one or more storage locations to save content. While storage can be managed at the infrastructure level, there is also a considerable amount of business level storage management that can occur in the application layer. The benefit of managing the storage at the application level as well as in the SAN, is in the ability to apply business logic to the content.

Content Engine manages storage in terms of Storage Areas and Storage Policies. Content is routed to Storage Areas (or a policy) on the basis of classes and folders. This framework allows the administrator great flexibility to guide how content is stored and ensure it is saved to the most appropriate storage medium. Content Engine also manages pools of storage that do not directly contain content, but are used to provide capabilities such as caching and indexing.

A content object exists in as many as three distinct pools of data. An object’s properties, which are stored in the database, include properties of the class the object belongs to and system data used to manage the objects state. Collectively, this is the object metadata, which is always stored in the object store database. The content of an object is stored in a storage area that can also be in the database, but is more commonly stored elsewhere such as a NAS for file system. The third pool is an objects index data. This data is created by an application packaged with Content Engine called Autonomy K2. The K2 application maintains an index of the text found in the content of an object store, which is maintained separately from the database and the content itself.

A single database can support multiple storage areas as illustrated by Figure 6-4. Storage areas do not have to be of the same type, as Content Engine is capable of addressing all of the storage areas independently. This allows the administrator to choose a naming scheme independent of the underlying storage.

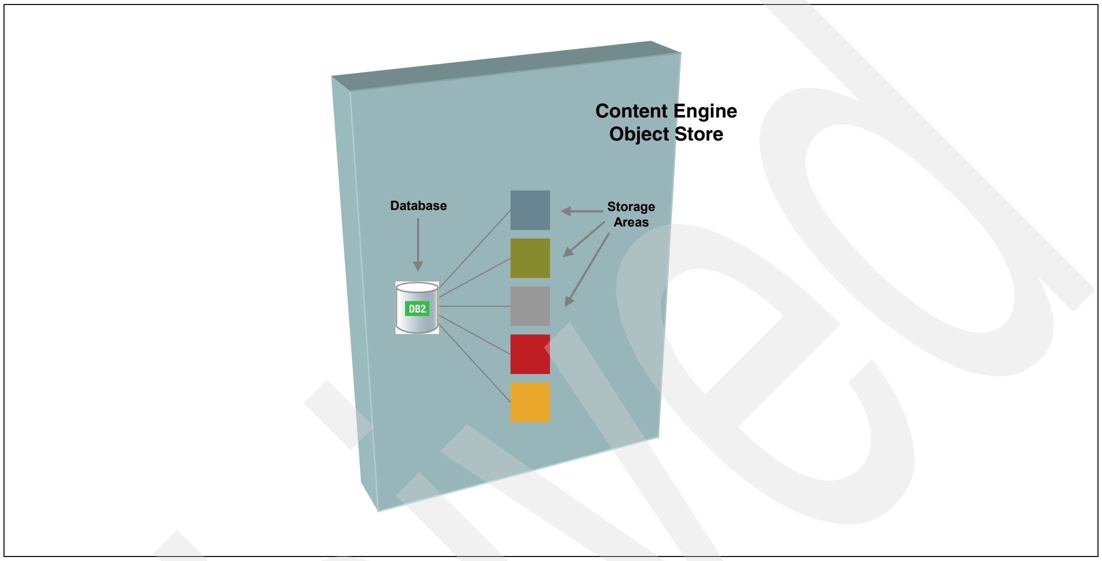


Figure 6-4 One to Many Relationship of Database to Storage Areas

For example, in a file system, directories help organize files into groups. These groups of files usually have a common relationship, they might belong to an application or a database. The files also have a relationship with files that are not in the same directory. Some might be word processor documents or executables. Content Engine provides the capability to assign different storage locations for files based on their business use without changing the directory structure.

In Figure 6-5 documents are stored based on membership to one of three classes. When users upload documents to Content Engine, they are prompted to specify the class the their document belongs to. Content Engine will then store the document according to the storage area specified by the class. Content Engine can be configured to pre-specify the class a document belongs to, allowing the administrator to manage document storage without requiring users to remember what class they must use to store documents.

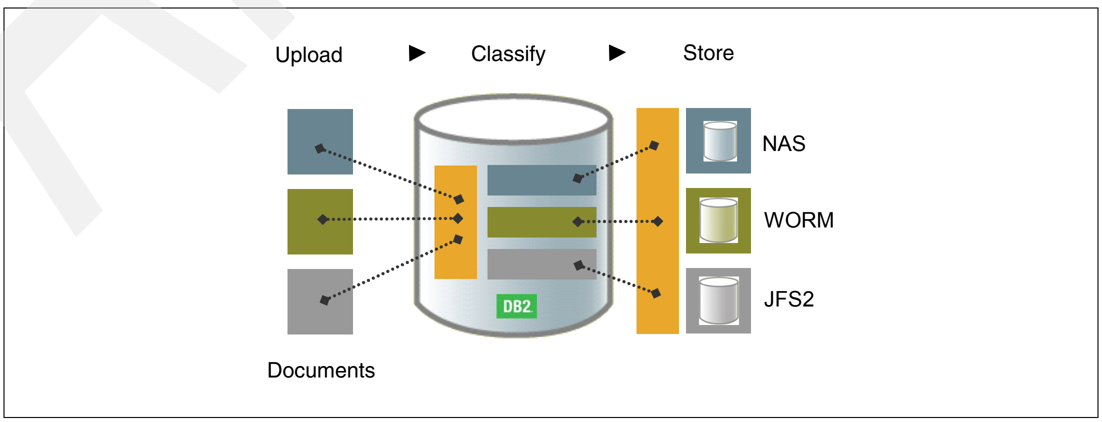


Figure 6-5 Class based storage routing

6.3.1 Storage areas

A storage area, as the name suggests, is any mechanism that is capable of storing files. This can be a file system supported by the operating system, a NAS device implementing CIFS (Windows platforms) or NFS (UNIX like platforms) or a supported Fixed Content Device (FCD). Different storage options have different capabilities and demonstrate strengths and weaknesses to be taken into account. The choice of storage area depends on greatly on how it will be used and the requirements of the business. More than one storage area might be required to ensure that content is stored in a manner is supports its business use.

Database storage area

As previously indicated, the core component of an object store is the database and is the default storage area for content. The document class or root class defines the default storage area for all content uploaded to the object store. Content stored in the database is saved in a blob format which, by convention, has an undefined upper boundary that would normally mark the end of a database field. Storing blob data hampers database performance because it defeats the database construct of fixed length fields. Fields of a fixed length allow the database to jump to predefined pointers containing the content of the row. Blob types on the other hand cause the database to scan the entire row to find the content.

The performance degradation caused by Blob types makes using a database storage area impractical. There are however exceptions. The first exception is for content automatically generated by the application. Site preferences and user preferences are stored as XML files in the object store and are instances of special system classes that defines how these files are handled. These files are critical to the proper operation of P8 and are preferably stored in the database, in their own object store. This ensures that users are able to log in to Workplace and other presentation applications even if a portion of the storage infrastructure is offline.

To illustrate this point, consider for a moment an object store with two storage areas, one is a WORM device, the other, a file system. If the site and user preferences were stored in the WORM device, the availability of that device would affect all of the object stores in the ECM environment. Additionally, any maintenance of this device will prevent users from utilizing P8. This configuration is especially troublesome when several departments share the ECM environment as it requires a maintenance window acceptable to every department when the WORM device is brought offline for maintenance. The same applies or the file system. The database, conversely, is always available, making it the ideal location to store preferences files.

The second exception is applicable when an object does not contain content. Content Engine allows objects to be created where a URI is given as a link to the content or without content at all. This is more of a rare exception but in these cases no storage area has to be created as no content is being uploaded.

File system storage area

A locally or SAN attached file system is the easiest storage area to configure and provides tremendous flexibility of implementation. This is achieved by relying on the operating system and storage hardware to provide a native volume on which to store data. The benefit of this being the ability to optimize the storage for the type and use pattern of the data.

When a file system storage area is configured, a pre-defined directory structure is created and managed automatically by Content Engine. This structure consists of twenty three numbered directories each containing twenty three sub directories. As content is routed to the storage area, files are evenly distributed amongst the directories ensuring no one directory receives the entirety of the newly uploaded content. This ensures that the directory structure can be recursed without causing prolonged pauses where the operating system is busy fetching inode (or Master File Table record on Microsoft platforms) metadata from the file system.

Recursion is a term used in programming to describe the process of calling a subroutine or method that subsequently calls itself until a specified task is completed. When a directory is recursed, the file system must return all of the permissions and time stamps of the objects in that directory. When there are many objects in a directory, the system call that does this prevents the program from continuing until the operation is complete. This behavior is also known as blocking I/O and is uninterruptable by the program because it is a resource handled by the kernel (user programs are typically unable interrupt kernel functions by design). To the user it appears as if the computer is "hung." Several tools depend on the recursion optimized structure described above, one of which is a utility to cross check the content with the metadata and ensure that both are in sync, hence the necessity for the predefined directory tree.

For each object in the object store, a corresponding file exists in the directory structure. On upload, Content Engine creates a globally unique ID or GUID to index the content and stores the GUID as part of the objects metadata. The GUID of the object, and its extension, are used to name the file that provides a point of reference to its associated metadata in the database. The path to the file is recorded in the database for future retrieval.

Network Attached Storage Area

Content Engine supports and NFS or CIFS capable NAS device and allows multiple Content Engine servers to act on the content at the same time. This capability allows content engine to be scaled horizontally and add capacity as it is required rather than purchasing a single larger server up front. Apart from the underlying transport and the ability to scale horizontally, a storage area on NAS is no different than one implemented on SAN or local attached storage.

Using NAS storage provides benefits not found in other storage implementations. Many NAS products have the capability of sharing over protocols such as HTTP. This provides the capability to publish URLs that point directly to content, bypassing the ECM software. This is of particular importance for organizations running high volume Internet facing applications. Bypassing the ECM software for Internet users allows an organization make best use of the hardware, because many Internet users might not require any of the ECM features. For example, an institution who publishes content consumed by general public would enjoy a considerable performance benefit by storing content to an object store whose storage area is a NAS device supporting HTTP. Internet users would access content without having to navigate the ECM software and the organization would not have to manage the performance of the entire ECM stack. An HTTP NAS configuration also opens up possibilities such as geographic caching and numerous other opportunities appropriate for delivering high demand content over the Internet.

Fixed Content Device Storage Area

An FCD is an implementation specific construct for supporting WORM storage. It is most commonly used to enforce specific behaviors on the content such as retention and restricting modification. WORM storage is used to ensure files on WORM disk cannot be modified or deleted until an expiration data has passed. The benefit of this capability lies in the ability to certify to third parties that sensitive records cannot be fraudulently modified and provide an audit trail as evidence. WORM storage is most typically applied to meet regulatory requirements. An audit trail allows auditors to review and certify on behalf of an organization the validity of the records they maintain in the ECM environment.

The FCD is also the how an Image Services environment is federated in P8. Image Services is different from WORM devices because it does not provide the records retention features of WORM by itself. It can however be used as a federation layer with its own storage on WORM devices. When used with Content Federation Services, one or many IS environments can replicate their index data to P8 and provide a single searchable catalog of the content.

From a programatic perspective, Content Engine connects to an FCD using a vendor API or leveraging existing capabilities of the underlying device. IBM P8 ships with the connectors required to read and write to several brands and models of WORM devices, the choice of which is dependent on the specific requirements of the organization.

IBM N Series SnapLock

IBM SnapLock is a feature of IBM N Series NAS storage. SnapLock is implemented from a storage area perspective in very much the same manner as a NAS storage area. SnapLock devices are set up as Fixed Content Devices in FileNet Enterprise Manager using the Add Storage Area wizard, which prompts for all of the configuration information required to connect to the device.

The FCD wizard creates a file structure the same as it would for a regular NAS device. When a file has to be retained, Content Engine sets the access time for the file to a time in the future and marks the file read-only. Until the new access time is reached, any attempt to write to the file is denied. These operations can also be executed in the OnTAP operating system. In OnTAP, the atime or access time of a file can be changed using the **`touch`** command and the permissions of a file can be changed using the **`chmod`** command. This capability has a distinct advantage over API driven WORM devices as it allows a NAS administrator greater visibility into the management of the NAS and implement retention on the NAS after the fact (the state of the content will not be reflected in P8 however). A NAS administrator can then immediately implement retention on the content without migrating to a new storage platform.

A typical ECM solution can command hundreds of terabytes of storage, usually as small files. Migrating large amounts data comprised of small files is a time consuming operation due to the performance implications of the file descriptor. A file descriptor is a special interface to a file that allows a process to read and write to it without the process having to know where all the blocks of the file exist on disk. As the descriptor is opening, no data is transferred and although the open() call is not computationally expensive on its own, the cumulative time spent opening millions descriptors greatly exceeds the time spent reading and writing the actual data. As such, the process of copying millions of small files to an another device requires first obtaining a file desrciptor for every file before transferring the data in the file, and thus given the amount of time wasted opening the descriptor, copying terabytes worth of data would require weeks or even months to complete.

Retention requirements today might not reflect retention requirements in the future and as industry regulations change, organizations might be required to begin enforcing retention. Migrating a large amount of data is costly and time prohibitive making a NAS solution capable of enforcing retention rules after the fact, a high value insurance policy. IBM N Series NAS devices can be brought into compliance without disturbing the existing file plan by applying a SnapLock license, and using a script to apply a retention date. Although this method would not be able to take advantage of the ECM records management features, the files would be in compliance to the satisfaction of an auditor.

IBM Image Manager Active Edition

Image Manager Active Edition (IMAE) is the product name used to describe an Image Services environment when it is federated by P8. Image Services in itself is an ECM platform optimized for image storage. Content is stored in Image Services in a similar fashion to Content Engine where the content metadata and the image data are stored in separate mediums. The metadata is stored in a database just as it is in Content Engine, but the image data itself has a broader number of options for archival.

Image Manager Active Edition (IMAE) can be used to store WORM data, but in an API like fashion. There is no direct file system interface which can be used to manipulate the data. Image Services supports several storage mediums and as the name suggests each is optimized for images. This makes the platform ideal for managing digital representations of paper documents.

- OSAR - Optical Storage Archive and Retrieval
- MSAR - Magnetic Storage Archive and Retrieval
- SSAR - SnapLock Storage Archive and Retrieval
- ISAR - IBM Storage Archive and Retrieval (DR550)

OSAR

OSAR was the original storage mechanism for Image Services data. A logical representation of physical OSAR surfaces manages where images are written in an optical library containing numerous optical storage cartridges, each cartridge containing one or more surfaces. OSAR is a write once technology and collectively with Image Services, it satisfies many regulatory requirements for auditable WORM.

MSAR

MSAR allows images to be stored on a file system instead of an optical disk allowing greater flexibility and faster retrieval of image data. MSAR can be implemented on any file system supported by the operating system, but MSAR data still implements the concept of surfaces. An MSAR surface is a container holding many images, which allows Image Services to overcome performance bottlenecks associated with storing small files. As files are added to a surface, images are copied back to back in a container. When the container size limit is reached, a new container must be used to store the next set of images. While MSAR does have a performance advantage over OSAR, it is not considered a WORM device because the individual surface files are modifiable.

SSAR

SSAR, using single document storage, is a best of breed storage solution for organizations seeking the performance of disk with the retention capabilities of OSAR. SSAR is implemented on IBM N Series hardware running SnapLock and functions in much the same manner as an FCD on Content Engine.

ISAR

ISAR, like SSAR, also uses single document storage. Figure 6-6 demonstrates how additional ISAR capabilities are brought to Image Services by allowing image data to be stored in a storage hierarchy. ISAR provides an intelligent storage framework for storing vast amounts of data on the most cost appropriate platform. The most frequently used image data can be stored on Fiber Channel, while less frequently used images might be saved to SATA. Images requiring long term archival can be archived to tape. Regardless of where the image is stored, ISAR provides a single contiguous name space for accessing the images. ISAR also has the same retention capabilities as SSAR making it the ideal platform for long term storage of infrequently accessed images. The migration of images between storage tiers is completely transparent to Image Services users.

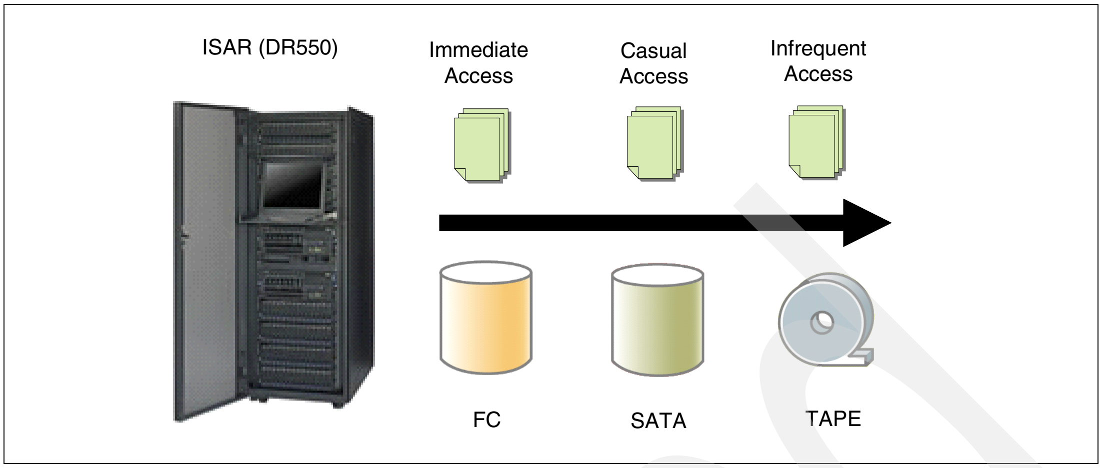


Figure 6-6 ISAR Hierarchy Storage Management for WORM

6.3.2 Storage policies

A storage policy is a logical group of storage areas. Every document class can be configured to store documents by storage area or storage policy. The benefit of a storage policy is the ability to load balance across multiple storage areas and by using storage policies, operating system and I/O bottlenecks such as recursion and I/O queuing can be avoided or even implemented after the system has been in production. These optimizations can greatly improve index creation and backup performance.

A storage policy helps mitigate performance problems caused by the growth rate of the content file system. By spreading the growth over multiple storage areas, fewer files exist per directory. Just as the Content Engine creates a directory structure to optimize for recursion in a storage area, the same strategy is applicable to storage policies. A single storage area contains a finite number of directories. Although the storage area will allow for millions of files, it is possible to eclipse the capabilities of the operating system as many exhibit performance problems when the number or objects per directory becomes too large. Creating multiple like configured storage areas assigned to a single storage policy allows the storage file system to scale considerably better.

6.3.3 Index areas

ECM users expect the search behavior of P8 to be similar to that of an Internet search engine and expect the language inside the file to be searchable along with the metadata. Indexes for Content Based Retrieval are stored independent of the content, allowing the index to be stored where it can be optimally tuned. This requires considering how often the index will be used. For example, an index might be necessary, but might be accessed on an infrequent basis. In this case the index might not require high speed storage. Cost is a serious consideration with regard to indexes, due to the rate at which they grow. Every word, phrase, and sentence in an indexible document format will be written to the index, causing the index to grow at a rate proportional to the content.

An Index Area is configured using the path name where the full text indexing information is stored. Multiple index areas can be used to hold data for a single object store, but a single index area holds data for only one object store. A single index area has only one file system directory, so multiple index areas are used for a single object store to spread the full text indexing information across multiple file systems, either for geographic location or scalability purposes.[10]

The use of multiple indexes allows the index functions of Content Engine to take advantage of multiple IO queues. From an operating system standpoint the kernel brokers read and write requests between applications and disk. Each file system has a queue where reads and writes wait to be handled by the kernel and requests are generally dispatched in a first in first out order. The benefit of this queue system is the ability to influence the scheduling of IO requests.

Figure 6-7 is an example of illustrating how a single queue compares to multiple queues when submitting multiple index requests. Each colored block represents a call to the read function of the operating system. In the single queue model two read requests are submitted the first completing in three steps. The second request does not complete for another five or eight total steps. This means the user waiting for Read A to complete must wait for eight IO operations before the data is returned. The two queue model executes IO in parallel. In this model even if each queue is polled one at a time, each request only has to wait five and six IO operations respectively before the reads are returned. The benefit to this model is a better balance of user access than in the single queue model where Read B is handled nearly right away, while Read A waits for other IO to be processed. Content Engine Indexes deliver a better balance of performance to all users by making use of multiple queues.

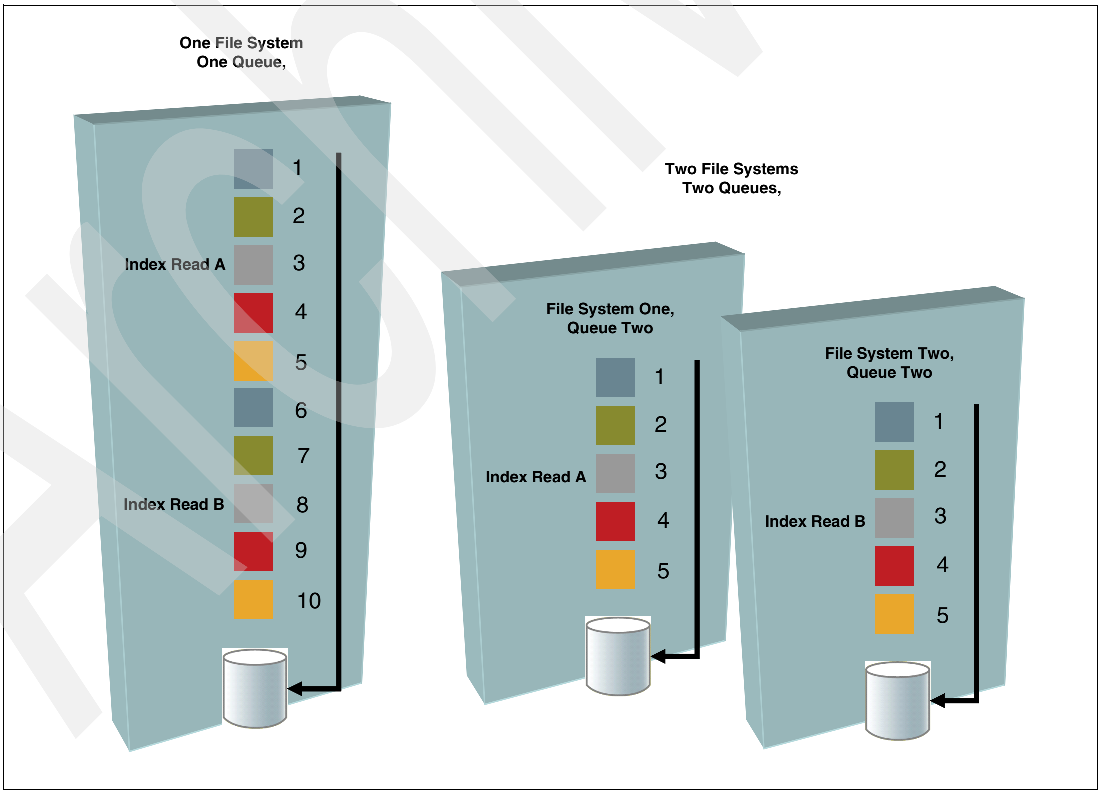


Figure 6-7 Single Queue Read versus Multiple Queue Read

10 IBM FileNet ECM Help

6.4 Content caching

Content Engine is capable of caching an object store using a content cache area. Content cache areas provide storage for frequently accessed documents accessible over a WAN. These content cache areas allow users that are geographically remote from the file storage areas to quickly access frequently requested document content.[11] While bandwidth over a WAN has improved dramatically in recent years, latency continues to hamper the overall throughput of highly interactive application such as ECM. A user begins to notice the impact of latency when it approaches 200ms for a round trip to the destination. While an individual is unlikely to observe this during a single transactions, many round trips as is typical in the ECM world will be clearly observable.

Organizations having branch offices stand to benefit most from content caching. Figure 6-8 illustrates how an ECM application in New York would be cached by a Cache Storage Area in London. When a user in London requests an object the request is routed to the local Content Engine. The Content Engine in London, not having the document, would request it from the New York Content Engine and store it in cache. Subsequent access to the cached document would occur locally thus requiring fewer round trips over the WAN overall.

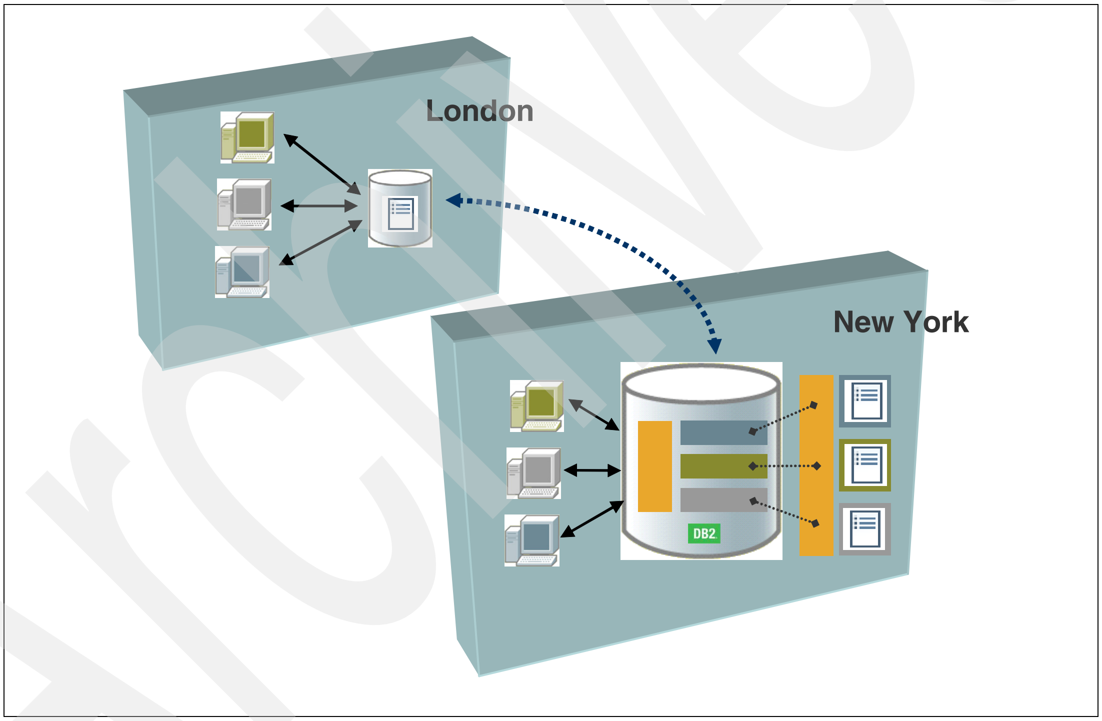


Figure 6-8 Overcoming WAN Latency Using Content Caching

Given the example above, the choice of storage plays a role in the user experience in London. Consider the choice of storage in terms of the number of users and their frequency of access. A Content Cache Area can be populated as it is used or populated when it is created. The advantage of pre-populating the cache is avoiding the preliminary impact of the user downloading content. If a user requires many documents and does not have to re-access them, the benefit of pre-population is considerable.

[11] IBM FileNet ECM Help

Another application of content caching is improving performance of heavily loaded WORM devices. A WORM device typically is used to archive large amounts of data making the cost per gigabyte an important factor when choosing a storage solution. SATA disk is usually chosen for this type of storage because of its price point relative to fibre channel disk at the expense of SATA's relatively poor performance on random IO. A Content Engine cache area using a small amount of Fiber Channel disk alleviates the problem by caching the data. This cache is also tunable so that the life span of the cache is in line with the time the objects are most actively accessed.

Figure 6-9 demonstrates how two WORM storage areas are cached by a Storage Cache Area. This configuration also allows the WORM storage areas (which can reside on different Content Engine Servers) to be used for heavy document injection. The users retrieve the documents from the cache instead of going directly to WORM for every document download.

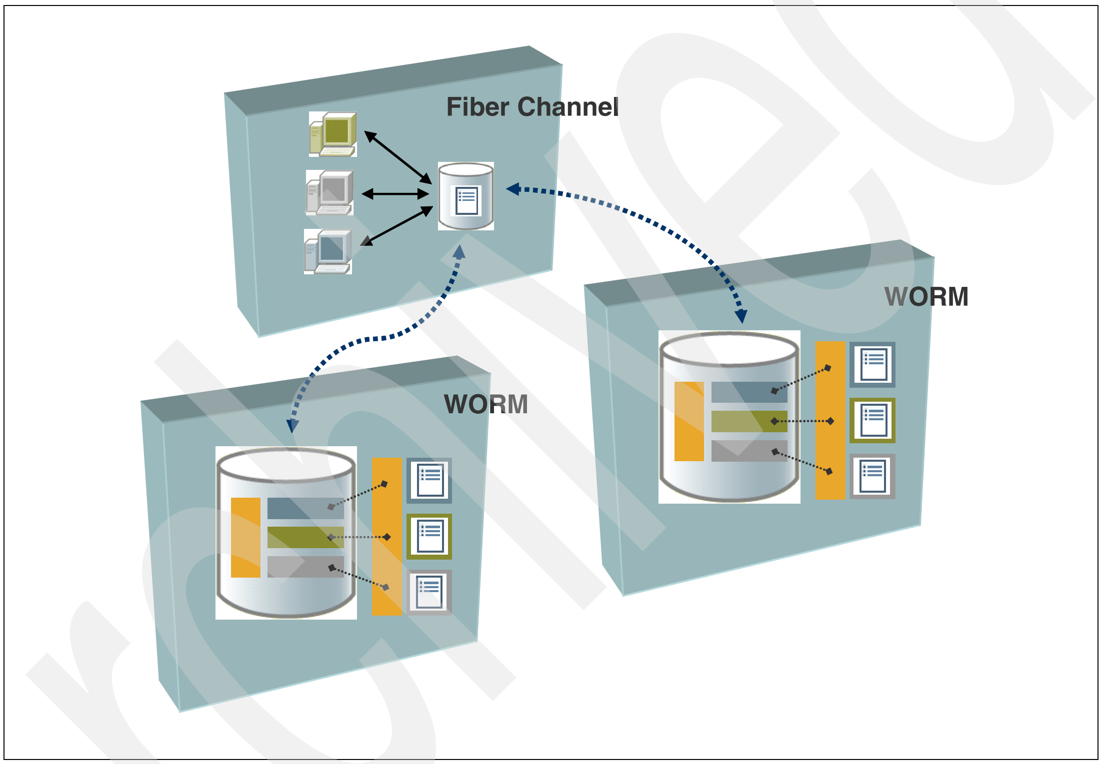


Figure 6-9 Using a Content Cache Area to Improve Storage Performance

6.5 Storage considerations for application recovery

An IBM FileNet ECM solution, involving the only the core P8 components, stores data on a number different physical volumes, managed by separate systems. The advantage of this is the ability to optimize the storage for the workload expected, but this flexibility is gained at the expense of complicating the manner in which the environment is backed up.

Figure 6-10 illustrates the data silos of a basic P8 environment that collectively is the mutable data required to run the ECM application. These silos contain the objects and workflows that make P8 useful to the business. Under normal circumstances, all of this data exists at the same point in time, meaning the referential integrity of the business data in each silo is maintained. If any one silo of data is out of sync with the rest, the integrity of the data from a business standpoint is questionable.

To better illustrate the importance of referential integrity, assume for that a P8 environment is used to process wire transfers for a bank. The process and content databases would have to maintain references between the content and the state of the workflow to ensure that collectively they are an accurate reflection of the customer account. If any of these components are out of sync, any wire transfers in progress would be in error. Given the liability of loosing or duplicating a transfer of money, any disruption or anomaly in this process could have dire consequences for the institution entrusted to execute the transaction.

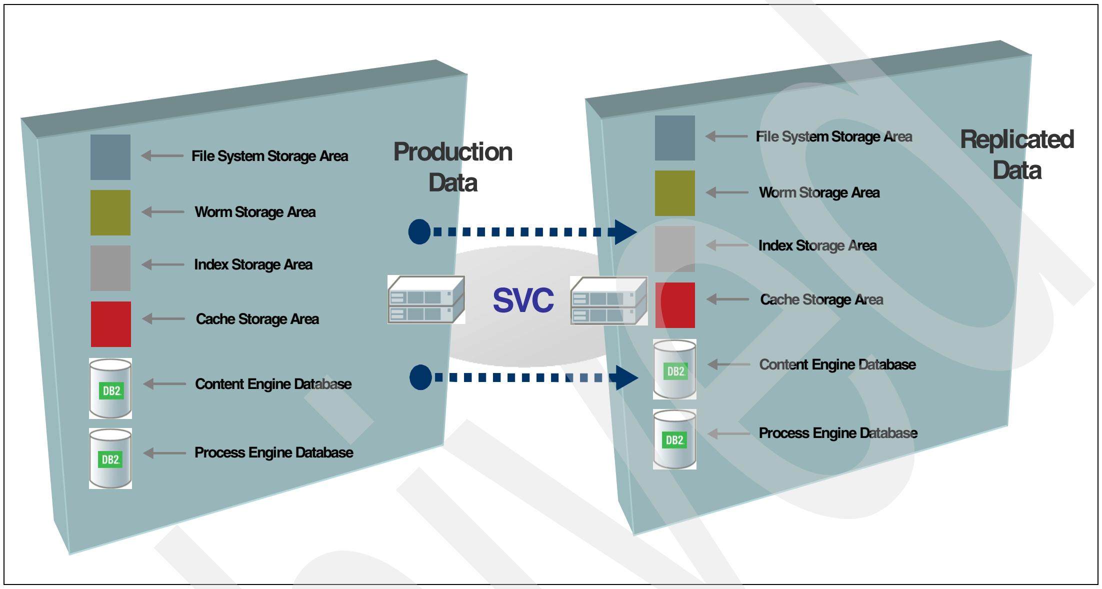


Figure 6-10 Replication Consistency Using SAN Volume Controller

It is therefore critical that when an ECM application is backed up or replicated, that all of the data silos are referentially accurate. This achieved by ensuring the replication strategy maintains write order fidelity across P8 applications, and backups occur while all of the data is in an immutable state.

6.5.1 P8 Disaster Recovery using tape

For P8, backup and recovery using tape provides a baseline of data protection. Tape backup requires using a specialized piece of software that connects to a server and copies its data to tape at regular intervals. In the event of a disaster, the infrastructure is re-implemented, P8 applications are re-installed, and relevant data is restored from tape. Tape is the most cost effective manner to provide disaster recovery for P8, however it is limited in the following ways:

- Data copied to tape must be in an immutable state for the duration of the backup procedure, requiring P8 to be shut down while backups are running.
- Tape demonstrates poor performance when used to back up many small files, which is typical for ECM applications.
- Availability of the data is dependent upon successful completion of backup jobs. Backup software must be carefully managed to ensure everything is captured correctly.
- Tape backup requires a dedicated administrator to ensure backups complete and media is shipped off site.

These limitations, in part, can be mitigated by combining the use of offline backups, logical volume snapshots, and other creative backup strategies such is synthetic backups. As effective as these tools are, tape backup always suffers from being human dependent. As such relying tape backup is best suited for organizations who can accept large periods of downtime to perform the back up, have generous windows of time to restore P8, and can withstand a loss of data as great as the time since the last successful backup.

IBM FileNet P8 stores data using multiple technologies such as file systems, databases, NAS devices, raw logical disk, and WORM devices. Each storage system usually requires a different backup technique. A database, for instance, might require tables to be exported to disk before a file system backup retrieves it, whereas NAS devices are typically backed up over NDMP, a specialized backup protocol developed for NAS and WORM devices. As with most ECM solutions, P8 uses a system of pointers to refer to the physical location of files and database records causing data maintained by each application component to maintain referential accuracy with other application components. When a P8 application no longer conforms to these relationships, errors occur, some of which might be difficult to resolve if at all. Ideally a backup strategy has to maintain these relationships and complete the backup of the environment while all the data is in an immutable state. Achieving this by shutting down the entire environment might cause the application to be offline for an unacceptable period of time due to the performance characteristics of tape.

To achieve acceptable throughput to tape, data must be sent to the tape device continuously while the tape head applies a magnetic signature to the media. An interruption in the flow of data requires the tape to stop and wait for more data to be sent. This characteristic is analogous to audio tape. Blank tapes were once used to copy music and a device with two tape wells allowed content to be transferred from one tape to another. Creating a tape with a mix of content involved exchanging the original tape for each new song. When a song ended, the recording would be paused until the next song was queued up. While the transfer might take three to four minutes, the process of queuing the two tapes could easily take longer. The process of recording data is similar, any interruption, such as queuing up a small file, will prolong the completion of a backup job.

Each application in a P8 environment is backed up differently, presenting different challenges from storage management standpoint. The data contained in these structures might require each to be referentially accurate and an out of sync record in a single system could pose significant fiduciary liabilities. This referential integrity becomes especially important when workflow or external systems of record are used. In these cases the referential accuracy of data can extend beyond P8 application components because the aggregate of the information represents the actual business record. These relationships might not be immediately obvious to backup administrators or even business analysts. Nonetheless, for the backup to be viable for restore, these relationships must be maintained.

> **Note:** Backup software generally does an insufficient job of capturing maintenance of the operating system. As operating systems are installed and patched, system images must be created to preserve known working environments. This prevents the practice of applying customization and patches required to run application components. These images must be part of the overall availability plan. Doing so negates having to manually re-create the server configuration from scratch, which might include installing numerous patches before a restore can be attempted.

6.5.2 P8 Disaster Recovery using replication

Data replication is the core component to disaster recovery. The risk to proprietary enterprise data is mitigated by replicating business data to an alternate location as users conduct normal operations. As users commit data to P8 it is written to a storage system at an alternate site. In the event of a disaster, the duplicated data allows application users to resume work using the duplicate copy. Data duplicated in this manner is suitable for use with P8 applications. As with any replication technology, there are caveats.

IBM FileNet P8 is a multi-tiered application that writes changes to a file system or fixed content device, a number of databases, and a small set of internal data files. As such, all of this data must be kept in a referentially valid state. In other words all of the data files, from an application standpoint, must be at the same point in time in order for objects to be valid from a business perspective in the replicated copy.

As with all capital expenditures, a replication strategy must be based in an assessment of risk versus the cost to mitigate the risk. Organizations that require tight recovery time objectives and recovery point objectives must consider lower risk replication technologies and periodically take file system snap shots while IBM FileNet application components are halted. Organizations willing to accept more risk can sacrifice some or all of the techniques to achieve a more amenable cost benefit ratio.

The biggest challenge of data replication is maintaining the write order of the data across multiple volumes and multiple systems. For those organizations who rely on the transactional integrity of the data across all aspects of P8, and possibly with other applications there is minimal tolerance for inaccuracies predicated by out of sync data. Replicated data is rarely transmitted over a distance where the replication can occur synchronously. Usually distances over one thousand feet introduce latency making synchronous replication invasive to the operation of the application. Thus replication usually occurs asynchronously where the replicated copy is an indeterminate number of minutes or even hours behind the active production copy.

The asynchronous nature of data replication makes the possibility of documents, metadata, and workflow being in sync highly improbable. IBM storage as part of an Information Infrastructure resolves this issue using SAN Volume Controller. SAN Volume Controller is a standards based SAN appliance capable of attaching to nearly any storage vendors SAN infrastructure and is implemented as a management layer between SAN attached storage and any number of hosts. This central point of administration allows all of the volumes in the SAN infrastructure to be replicated as a group and maintain the write order of all of the data regardless of the host accessing it. Because data is replicated a group, write order is maintained for all of the hosts who wrote the data.

7

IBM Content Manager and IBM Content Manager OnDemand: Overview

In this chapter, we take an introductory look at IBM Content Manager (CM) and IBM Content Manager OnDemand (CMOD). We show how the products can utilize the IBM Information Infrastructure for the long term secure storage of data.

We focus on the differences in the products, introduce their possible uses, and explain how these uses could impact your storage requirements.

In the later case studies, we look at specific implementations of these products in a real world scenario.

7.1 IBM Content Manager

IBM Content Manager (CM) is a scalable and secure content management platform that is content agnostic. That is, it can be used to store any type of digital asset including, but not limited to, multimedia files, workgroup documents, e-mails, and images. The primary use for Content Manager is for the storage of individual digital assets.

A specific use would be e-mail archiving using products such as IBM DB2 CommonStore for Lotus® Domino® (CSLD) or IBM DB2 CommonStore for Exchange (CSX). Both of these products, as well as IBM DB2 Document Manager, can utilize CM as their repository. We have included an e-mail archiving case study later in this book.

CM has tight integration with IBM DB2 Records Manager (RM) and protected storage that allows CM to be used as a compliance solution for the digital corporate records of enterprises.

CM provides a comprehensive API that allows its capabilities to be integrated into and utilized by partner and end-user written applications.

For more information on Content Manager, see the book *Content Manager Implementation and Migration Cookbook*, SG24-7051.

7.1.1 Benefits of IBM Content Manager

These are the key benefits of Content Manager:

- Provides an extremely rich metadata framework that allows the solution architect to extensively model document attributes and relationships
- Allows for compliance by utilizing electronic records management when integrating with Records Manager
- Offers a flexible security model that allows the solution architect fine grained access control over all objects in the system, including documents and workflows
- Allows for the full text indexing of stored documents
- Provides workflow using the built-in Document Routing capability
- Includes an extensive API to allow for third party integration
- Can exploit the storage management capabilities of the IBM Information Infrastructure

7.1.2 Architecture

Content Manager is a loosely coupled distributed system with two main components:

- Library Server
- Resource Manager(s)

In the following sections, we discuss these components.

Content Manager Library Server

A Content Manager system can have only one Library Server and it is the central repository for all metadata, configuration, and security in the system. The Library Server is installed within an RDBMS, for example, DB2 Enterprise Server Edition, and runs as a number of stored procedures within the database.

Client applications communicate with the Library Server using the Content Manager API over a database connection, for example, DB2 Run Time Client.

Storage is required for the database executables, the transaction logs, and the database tablespaces.

The database administrator is able to modify the layout of the database to suit the underlying storage by modifying the IBM supplied database definition scripts. For example, the DBA could decide to use Database Managed Space (DMS) rather than the default System Managed Space (SMS) for the layout of the various database containers on disk.

Content Manager Resource Manager

Resource Managers are independent storage nodes on the network that understand how to store objects to the underlying storage system. There must be at least one Resource Manager within a CM system and there can be up to 255.

The CM administrator can dictate which Resource Manager an item must be stored in and, if replication between Resource Managers is enabled in the system, to which other Resource Managers the data must be replicated. Each Resource Manager can have its own storage configuration, and each different document type can be assigned to different storage locations if there is a requirement for separation of data at the storage level.

The Content Manager administrator also works with the storage administrators to assign objects to specific storage devices either at the local disk level (DASD, SAN, or NAS), called LBOSDATA, which means “LAN based object store data”, or via Tivoli Storage Manager.

Each Resource Manager will also require storage space for the database executables, transaction logs, and for the database tablespaces.

Additional Resource Managers can be added to the system at any time. Additional Resource Managers can improve redundancy so that the failure of an individual Resource Manager can be tolerated. Additional Resource Managers can also lessen the impact of slow WAN links between, for example, regional offices, as the additional Resource Manager can, through replication and object caching (LAN caching), make documents available locally.

Resource Manager replication allows the administrator to proactively copy data to another Resource Manager. This is useful for disaster recovery purposes, as Content Manager can serve object data from the secondary copies of the objects is the primary copy is unavailable due to the primary Resource Manager being down.

LAN caching is a form of adhoc replication. An object is copied to another Resource Manager at the time of retrieval and stored in the cache area. This allows the object to be retrieved from the second Resource Manager at full LAN speed.

Figure 7-1 shows the architecture of Content Manager and the storage that is required to underpin a Content Manager solution. Additional Resource Managers are not shown in this diagram.

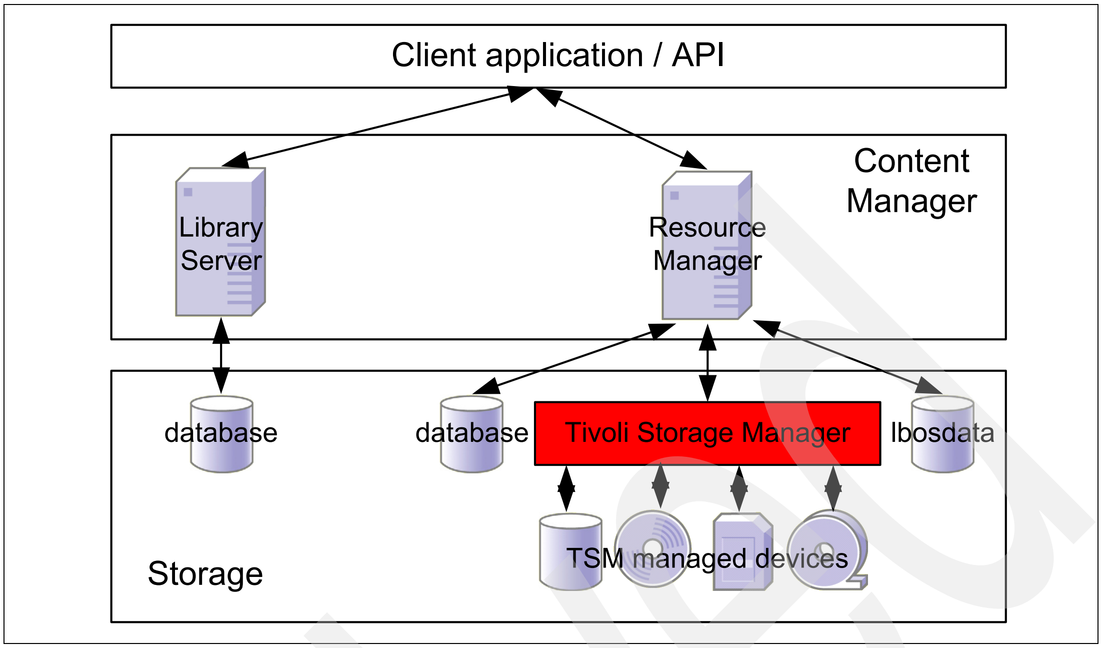


Figure 7-1 Content Manager architecture

7.1.3 Access control

Content Manager has a number of different levels of security that control many aspects of access to the system and content.

- Privileges:

  Privileges are the most basic functions that can be performed in Content Manager. There are approximately 70 privileges that are assigned to users and access control lists through Privilege Groups and Privilege Sets. Examples of privileges include these:

  – **ItemAdd:** This is a user allowed to import objects.

  – **ItemQuery:** This is a user allowed to search for objects.

- Privilege Groups:

  By themselves, privileges are not overly useful in Content Manager, because they only allow access to very specific operations. In all cases, to be able to perform meaningful work in Content Manager, a number of privileges will be required to perform a task.

  These collections of privileges are call Privilege Groups.

  For example, one of the standard Privilege Groups that is created as Content Manager in installed is called *ClientTaskCreate* and allows a user to import and scan documents, and to create items and attributes — that is, all the privileges that are required to create an object on the repository. This Privilege Group comprises four individual privileges:

  – ItemAdd
  – ClientScan
  – ClientImport
  – ClientAddNewBasePart

In essence, we are creating a *Role* that allows users to create content. We would create other *Roles* to perform other functions.

Other standard Privilege Groups include:

  - **ClientTaskDelete:** This allows a user to delete objects
  - **ClientTaskUpdate:** This allows users to update existing objects.

- Privilege Sets:

  A Privilege Set is a grouping of Privilege Groups and privileges that are assigned to users and to Access Control Lists. We are essentially indentifying the different roles that must be assigned to users in the system. For example, if we have users that are allowed to Create Objects, Delete Objects, and Update Objects, then we would create a Privilege Set that contain the ClientTaskCreate, ClientTaskDelete and ClientTaskUpdate Privilege Groups.

  Privilege Sets are assigned to each user that is created within Content Manager. The Privilege Set that is assigned is the absolute maximum set of privileges that a user is ever allowed to perform inside Content Manager. If the user's Privilege Set does not contain a privilege then the user is never able to perform that action regardless of any other security settings in the system.

- Access control lists (ACLs):

  Access control lists are assigned to objects within Content Manager and dictate which users are allowed to perform which actions against a individual object.

  An access control list is made up of a list of 2-tuples consisting of a username / groupname and a Privilege Set. That user / group then has the privileges as defined in the privilege set for the object the ACL is applied against.

  The ACL can never increase the privileges assigned to a user via the user's own Privilege Set. Therefore, if a user's privilege set does not have the Delete privilege, but the ACL does give the delete privilege to the user, the user is still not allowed to delete as the user's privileges do not allow the user to delete.

7.1.4 Records Management

Records Management enablement of Content Manager is through the use of the DB2 Records Manager component.

Records Manager is tightly integrated with Content Manager and allows for objects in Content Manager to be declared as records and classified into the file plan. After an object has been declared as a record, then Records Manager takes control of the object and will apply its own security to the objects so that they cannot be modified or deleted outside of the auditable records management process.

7.1.5 Other features of Content Manager

Content Manager provides a Document Routing capability that allows for business processes to be defined that objects can then be processed through.

For example, Figure 7-2 shows a simple insurance claim process that could be modelled within Content Manager. Users are assigned by the CM administrators to each step in the process, and when work is available in a particular step, the work items will show up on the users work lists.

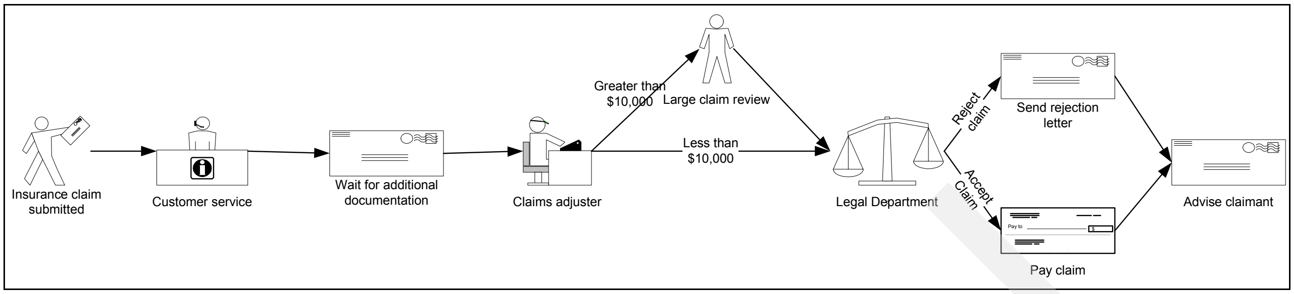


Figure 7-2 Simple insurance claim business process

Other capabilities within Content Manager include:

- Versioning:

  Versioning can be enabled to allow for tracking of meta data and objects if they are updated in the repository. Users that have the correct privileges within the system are able to look at past versions of documents.

- Annotations:

  Annotations allow authorized users to add markings like text, notices, lines over the top of an object. These annotations are then stored with the object in the Resource Manager and can be seen by authorized users. Note that Annotations do *not* change the original object, annotations are layers that are applied on top of the objects when required.

- Note logs:

  Note logs are a text log that can be attached to an object by authorized users and viewed by authorized users. Much like annotations, note logs do not change the object itself but rather is just stored with the object.

- User exits:

  User exits allow the behavior of the Content Manager system to be modified. One important user exit that is supplied with Content Manager is the LDAP user exit. This allows Content Manager users to be authenticated against an external LDAP database.

  Other user exits that can be implemented include these:

  – ICMFetchContentExit is used for processing objects after they are retrieved from storage but before it is returned to the client application.

  – ICMValidatePassword is used to ensure that passwords meet required length and complexity requirements.

  – Document Routing exits are used to allow steps in document routing processes to call external routines. The exits could be used to monitor for SLA issues.

  – ICMGenPrivExit and ICMACLPrivExit exits allow the standard Content Manager privilege and access control list behavior to be overridden.

Service Level Agreements (SLAs) can be monitored and escalated using in built functionality that will set a Notify flag when an SLA has been exceeded, along with information that could be obtained by using the Document Routing user exit.

7.1.6 Client applications

There are a number of client applications that are available for Content Manager. There is also a fully published API that allows third parties to develop applications to use Content Manager as a repository.

The IBM CommonStore family of products can use Content Manager as the repository for archived data.

CommonStore for Lotus Domino and CommonStore for Exchange are eMail archiving solutions and can store archived eMails into a Content Manager repository. eMails can also be declared as records if required.

CommonStore for SAP allows data and reports to be archived from an SAP system to reduce the database overhead of the SAP system.

DB2 Document Manager is a workgroup tool that allows group members to collaborate while authoring, revising and approving workgroup document.

Additionally, three Content Manager clients are shipped with the product.

- Windows Client

  This is a Microsoft Windows 32-bit application that allows the end user to:

  – Import objects
  – Search for objects
  – Reindex objects
  – Retrieve objects
  – Delete objects
  – Interact with Document Routing
  – Declare objects to Records Manager

- eClient

  eClient is a Web application that is installed into WebSphere® Application Server. This client gives the user much of the same functionality that the Windows Client can provide.

- Administrative Client

  The Content Manager Administrative Client is used by administrators to configure the system with function such as these:

  – Maintain users
  – Maintain metadata attributes
  – Maintain Item Types
  – Maintain Resource Managers and storage rules
  – Maintain Document Routing processes

  The administrative client is not able to view any metadata or content.

7.1.7 Data access patterns for an IBM Content Manager solution

A data access pattern looks at how digital assets are used during their lifecycle, from the moment it is stored into a content management system until its ultimate destruction. For some systems, the data access pattern is very well defined, but in other cases can vary depending upon the business process the data is used to support.

There is no standard data access pattern for a CM solution because Content Manager can be utilized in many diverse solutions.

As an example, next we look at a Workgroup Document Management System using DB2 Document Manager as the client application, and utilizing CM as the underlying repository. We determine its access patterns and how this solution could map onto the physical storage.

Example: Workgroup Document Management System

Consider a small workgroup in an enterprise whose primary job function is to create and revise corporate policy documents. These documents are authored and reviewed within the workgroup, and once approved, are declared as a corporate records and made available to other employees.

Business requirements

The workgroup has access to create, update, and approve policy documents. This access must be as fast as possible to allow for productive use of the workgroup's time.

After a document has been approved, it is converted to PDF format and is stored in the repository so that all users in the organization can view the document. At the same time, the original document is declared as a corporate record and classified into the correct location within the organization's file plan.

The original document will not be modified again, however, it can be viewed and copied on an irregular basis.

When, at some later stage, the policy document has to be updated, the above process begins again. A copy of the document that was last declared as a record is taken and this is the document that is then modified to create the new revision.

Use cases

Next we discuss some Workgroup Document Management use cases.

Document creation and modification

In this use case, a workgroup of authors, editors and approvers create, edit, and finally approve documents for publication.

Documents are small, possibly up to several Megabytes but they are retrieved and stored on a regular basis. To achieve the highest productivity, these documents must be stored on high speed storage.

Document rendition, declaration, and classification

After the document has been approved for publication, it is transformed into a PDF document and the original document is declared as a corporate record.

We therefore have two different storage requirements for the two different copies of the document:

- Original document:

  This document is locked down as a corporate record — it is not retrieved regularly and cannot be modified. It cannot be deleted until the organization's Records Manager allows the object to be destroyed.

- PDF rendition:

  The rendition of the document is not declared as a record and will be accessed regularly.

Rendition / End user usage

For the life of the PDF rendition, users must have timely access to the PDF version of the policy document.

Document revision

When a policy document has to be updated, a copy of the document that was declared as a corporate record is made and this new document then goes through the above use cases.

Overall access requirements

Looking at the access requirements, we can see that we have four distinct phases of access:

1. Initially we require high speed access to documents for creation and update.
2. After the document is finalized, it must be written to protected storage for storage of the corporate record. Speed of access is not really an issue as retrieval rate is very low.
3. Normal access to the document on a daily basis can be slower but must still have good response time, possibly < 1 second. As there are a large number of policy documents, they do not have to be stored long term on fast disk, but could be moved to slower, less expensive disk, while still allowing acceptable response times.
4. During creation and editing of a new revision, the new document is stored on high speed storage and further document access returns to phase 1.

Figure 7-3 shows these access requirements.

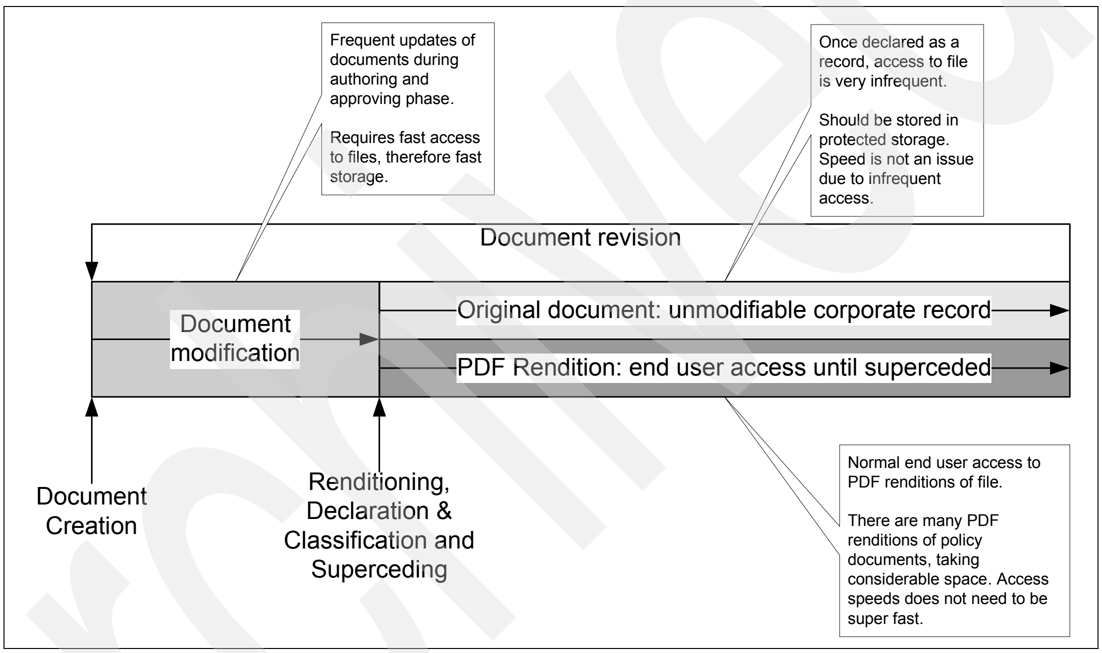


Figure 7-3 Storage lifecycle of documents

Mapping the requirements onto the storage

Taking the above access requirements into account, we can determine how to map the various storage requirements onto the storage subsystem.

In our example, we require one Library Server and one Resource Manager. We require a database for each, and these databases must reside on fast local storage.

For the storage of the objects, we require several different types of storage if we are to meet the stated storage requirements.

We will store newly created documents, including new revisions, on the LBOSDATA area, which normally resides on fast local disk.

For longer term storage of objects, we will utilize the capabilities of Tivoli Storage Manager to manage the storage of the corporate records and renditions; however, due to the different long term storage requirements, we will require two different types of storage within Tivoli Storage Manager. For the corporate records, we require retention management to ensure that the documents cannot be deleted or updated, and for the renditions, we just require standard storage.

Therefore, for the document management system, one possible solution for the storage for the system could be:

1. SAN or NAS — Library Server database — is used for life of the documents.
2. SAN or NAS — Resource Manager database — is used for life of the documents.
3. SAN or NAS — Resource Manager LBOSDATA storage — the documents stay here until declared as a corporate record.
4. Tivoli Storage Manager managed storage — the storage of PDFs rendition — remains on this storage until deleted.
5. Tivoli Storage Manager protected storage — the storage of corporate records — remains on this storage until record is allowed to be purged.

Note: The Resource Manager itself could have managed the storage of the PDF renditions if we had chosen to use slower disk storage. By using Tivoli Storage Manager, we are allowing for the use of other devices such as tape and optical storage.

7.2 IBM Content Manager OnDemand

Content Manger OnDemand (CMOD) allows for the archival and distribution of very large computer generated reports and data sets.

For example, in a large bank, a monthly credit card statement print run might generate hundreds of thousands, if not millions, of pages. Using CMOD, these reports can be indexed, stored on long term storage, and possibly declared as corporate records.

CMOD can store items in any format if the index information is provided at the time the item is stored. If the objects are of the following formats, indexing information can be extracted during loading or documents that been transformed using transformation technologies such as the Xenos d2e platform.

- Line data
- AFP
- PDF

CMOD also has the notion of a “Generic Index File” that allows CMOD to archive pre-indexed data. The Generic Index File would contain all of the meta data information about the objects to be stored so in essence CMOD can be used for the storage on any file types.

It is not uncommon for data to be stored in a CMOD system for 7, 10, or even 50 years. In some instances, data might effectively never expire and our underlying storage system has to support this retention requirement.

For more in-depth information about Content Manager On Demand, refer to the book, *Content Manager OnDemand Guide*, SG24-6915.

7.2.1 Benefits of IBM Content Manager OnDemand

Here are the key benefits of IBM Content Manager OnDemand:

- Metadata can be extracted from reports as they are loaded, or data can be pre-indexed by the generating application.
- Streamlined distribution of report content using the Report Distribution feature allows reports to be e-mailed rather than printed.
- It supports compliance by utilizing electronic records management.
- It allows for the transformation of print streams at time of archiving, such as Xerox Metacode or PCL to PDF.
- It can exploit the storage management capabilities of the IBM Information Infrastructure.

More information on IBM Content Manager OnDemand can be found at:

http://www-306.ibm.com/software/data/ondemand/mp/

The optional Content Manager OnDemand Report Distribution feature can automatically distribute report "bundles" to users using e-mail or it can print on the receivers printer. The CMOD administrator controls the reports that are included in the bundle and the destination of the bundle.

7.2.2 Architecture

Content Manager OnDemand is made up of two components:

- Library Server
- Object Server(s)

Content Manager OnDemand Library Server

The CMOD Library Server is equivalent to the Content Manager Library Server in that there can be only one per CMOD system and it is responsible for the storage of metadata, configuration and security.

The Library Server is an application that utilizes an RDBMS to store its configuration and data. Unlike Content Manager, CMOD does not utilize stored procedures within the database but rather runs as a standalone executable using the database for storage.

Content Manager OnDemand Object Server

The CMOD Object Server is, much like the Content Manager Resource Manager, the component of the system that manages the storage of the objects on disk and Tivoli Storage Manager.

There can be more than one Object Server in a CMOD environment.

> **Note:** The Content Manager OnDemand Library Server has the Object Server functionality embedded within it.
>
> In its simplest installation, the Content Manager OnDemand Library Server and Object Server are combined on one server. There is no necessity for a separate Object Server installation in a single server environment.

When configuring an Object Server, the administrator dictates the rules for the migration of content from disk based storage, for example, cache to Tivoli Storage Manager.

The local storage for objects is called the “cache” and it can be made up of a number of different file systems, but they are seen by CMOD as a single storage area. It is not a cache in the classic sense of the word, rather the cache refers to locally attached disk storage on the CMOD server and this is usually the first storage location for images or reports that are loaded into CMOD. When objects are retrieved from CMOD they are NOT placed back in the cache.

Unlike Content Manager, it is not possible to separate different types of document on the disk as there is only one common cache area in a CMOD system.

The Content Manager OnDemand Object Server also does not require an additional database, unlike the Content Manager Resource Manager, which does require its own database.

By defining a “Storage Set,” the administrator defines which Tivoli Storage Manager storage node objects can be migrated to. There can be multiple Storage Sets defined in the system, each of which its own TSM volumes.

The “Application Group” configuration dictates how long the objects will live on cache disk, and when migration occurs. Migration can be configured to occur:

- **Never:** This means that the object will only ever be stored on cache.
- **When data is loaded:** This means that the object will be stored to Tivoli Storage Manager at the same time that it is stored in the cache.
- **On next cache migration:** When the `arsmaint -m` command is run ALL objects will be migrated
- **After a certain number of days in the cache:** When the `arsmaint -m` command is run, all eligible objects will be migrated

The destination volume of a migration is defined by the storage set assigned to the Application Group.

End user access to data is via a CMOD Folder. The Folder is a view across several Application Groups.

Figure 7-4 shows the architecture of Content Manager OnDemand and the storage that is required to underpin a Content Manager OnDemand solution. Additional Object Servers are not shown on this diagram.

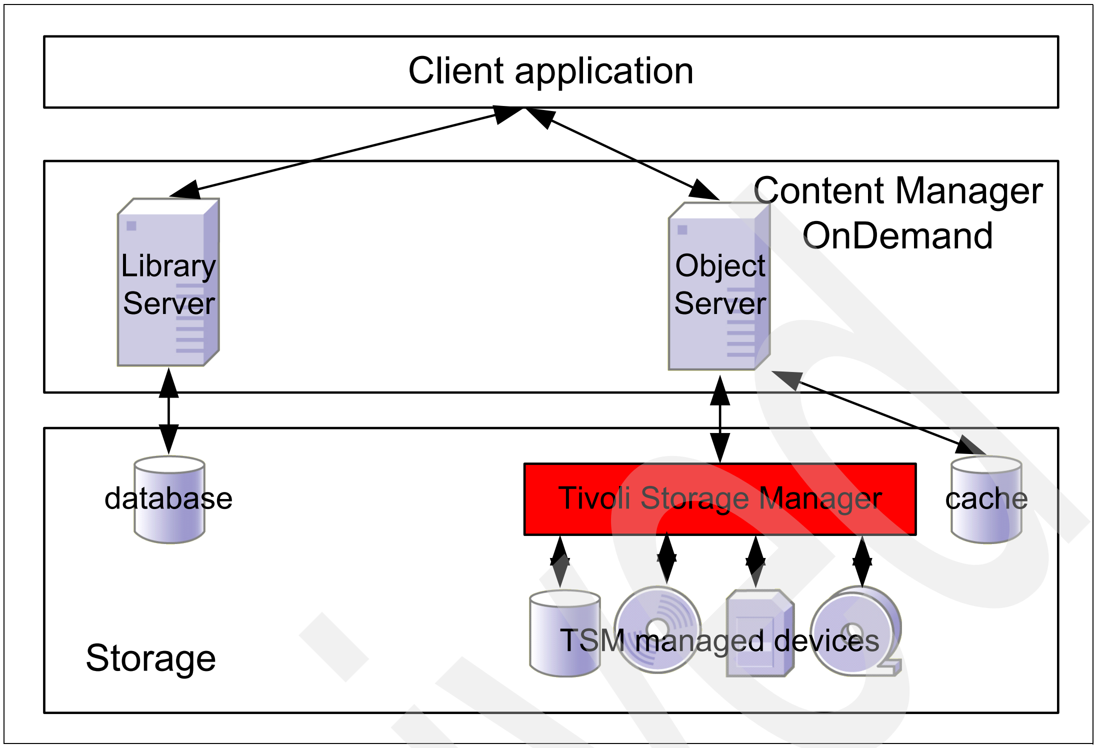


Figure 7-4 Content Manager OnDemand architecture

7.2.3 Access control

Compared to Content Manager, the access control in Content Manager OnDemand is far simpler.

Permission to data is given at two distinct levels. Each Application Group has permission information for the objects and for annotations, which consist of optional additional information that users can save about an object. CMOD can assign View, Add, Delete, Update, Print, Fax, and Copy permissions to CMOD users and groups.

Additionally, the Folder that straddles the Application Group(s) also has access control information dictating who has access to the Folder. If a user does not have access to a Folder, then they are not able to view data regardless of the permission on the underlying Application Groups.

Similarly, if a user has access to a Folder, but not the underlying Application Group, they will not have access to data in the Application Group.

7.2.4 Client applications

There are several CMOD client applications. These include Windows based applications for end user access and administration, as well as a number of Web based applications:

- WEBi:

  The IBM Web Interface for Content Management (IBM WEBi) is a standalone, out-of-the-box client that provides users with a Web console for working with content from multiple content servers, and for arranging that content according to your business requirements.

- OnDemand Web Enablement Kit (ODWEK):

  ODWEK allows users to access data that is stored in an IBM Content Manager OnDemand server by using a Web browser or a user-written program.

  ODWEK also supplies an API that allows an application to be written to communicate with CMOD.

- eClient:

  eClient is another Web application that allows end users to access objects stored in CMOD. eClient can also access objects stored in Content Manager.

Information Integrator for Content (II4C) provides an API that allows for third party applications to be built for CMOD, as does ODWEK.

7.2.5 Data access patterns for IBM Content Manager OnDemand solution

Typically, in a CMOD system, the volume of objects being stored is very large and the retrieval rates for data are usually quite low. For many applications, retrieval rates for data would be in the 1% to 10% range of ingested data. However, no two types of data or projects have the same data usage patterns. The solution architect will have to determine the exact access patterns for each individual project.

To illustrate a somewhat typical access pattern, we now look at a CMOD solution, a check archive with staff and end users accessing the images over the life of the check image.

Example: Check archive processing

Check processing involves large quantities, possibly several million, small images being stored into the archive on a nightly basis. Images are then viewed by staff and customers over the life of the check image in the archive. Most accesses to checks would occur in a very short time period after the check images were archived, typically days to weeks. Less frequent access to images would occur over the coming years.

Business requirements

Check processing occurs on a regimented time frame as the exchange of the check's value between the banks must occur before the overnight processing batch runs begin. This overnight processing determines exceptions that might require human intervention to be resolved the next day, which would utilize the check images stored on CMOD.

Staff would view checks that caused an error in the overnight processing the day after the check was imaged.

Going forward, customers, through the Internet banking portal have to be able to view check images sometimes years after the check has been honoured.

Because check images contain personal information that must be kept in confidence between the drawer and drawee, access to and storage of images must be secure.

If check truncation[1] has been implemented, then we must ensure that the storage system provides integrity and availability to allow the clearance and settlement process to be completed secure and in the required timeframes.

---

[1] Truncation is the process whereby checks are not physically moved from the depositors bank to the drawers bank. Images and pertinent information about the check are exchanged electronically. That is, the instrument is not physically exchanged.

Therefore, a solution and the storage must address these underlying requirements:

- Allow for timely ingestion of data
- Allow access to images by staff at the start of the business day after the images were loaded. Access speed will impact staff productivity therefore fast access to images is essential for newly archived images
- Allow customers to view images for many years into the future. Access speed is not of paramount importance as customers will be accessing the images over the Internet, which allows for a slower response time than the Local Area Network.
- Provide secure, trustworthy, and highly available storage of images
- Provide for a range of response times from very fast to slower, possibly offline as the age of an image increases

Use cases

There is a use case for the loading of images and several use cases for access to the images stored. They have different access patterns that must be considered when architecting the storage solution.

Data loading

Loading of the image data is one of the most important aspects of a check archive as we are required to load a large amount of data over a relatively small period of time.

For example, if we say that a single check image is a 15K TIFF image, and that we process 1 million checks each evening, we require a storage system that is capable of loading 15 GB over an eight hour period.

The loading process will be the most throughput intensive form of access to a CMOD system.

Day 2 processing staff access

"Day 2 processing" is the process of remedying the exceptions that have been identified during the overnight batch processing of the checks.

Possible exceptions include these:

- The account the check was drawn upon has insufficient funds and the check must be dishonored.
- The check's value might have been posted to a suspense account, as the actual account information on the check or credit slip might have been scanned or entered incorrectly.

As part of the processing of the exceptions, bank staff will usually view the images associated with each transaction. For example, for the second exception listed above, bank staff might post the value to the correct account after viewing the credit slip and updating the account details.

Importantly, the percentage of checks that cause an exception is very low, possibly in the order of only one percent, or less, which means that the overall retrieval rates by Day 2 processing staff is extremely low for new images and virtually non-existent for images older than two days.

Another characteristic of access by Day 2 processing staff is that access will usually occur on a LAN and staff would expect reasonable response time for viewing of the images.

Fraud detection staff access

Check fraud detection could require access to the check images throughout the life of the check image and access will, most likely, be random in nature. Given that retrieval rates would likely be quite low[2] it would not be unreasonable to say that these retrievals would be batch based and utilize system resource when loading and end user access is not occurring.

Customer self service access

Bank customers, on the other hand, if they have access to their own check images through the bank's Internet banking portal, would, most probably, only look at their images after the next bank statement was delivered in an effort to reconcile their check book.

Some customers might leave reconciliation until tax time, which means that they might be attempting to retrieve images that could be up to 18 months old.

Also note that customers will be accessing the check images through the Internet where response time is not as critical as on a LAN.

It might also be reasonable to say that access to very old images do not have to be immediate.

Overall access requirements

Based on the combined access requirements, we see a number of different "phases" of access that could be applied to the check images. These will dictate the different types of storage that can be used in the solution, as follows:

1. Day 1: Ingestion of check images

   Requires very fast storage for efficient ingest of the metadata and images. Throughput must allow for the storage of all images by the time "Day 2 processing" begins.

   No retrievals are performed at this time.

2. Day 2: Exception processing by bank staff

   Throughout the day, the check transactions that caused an exception will be viewed by bank staff and the exceptions cleared. Access to images must be as fast as possible to allow bank staff maximum productivity.

3. Day 3 to 30: Very occasional access

   Customers can view recent check images.

4. Day 30 to 550: Normal access to images

   During this time frame, we will have occasional access to images by customers and bank fraud staff. Most people would expect reasonable access speed to images.

5. Day 550 onwards: Occasional access

   It would be reasonable to expect slower access to images over 18 months old, possibly even batch retrieval of images that are made available after 24 hours.

[2] Again, your solution architect would have to confirm this assumption while gathering the project requirements

Figure 7-5 shows these access requirements.

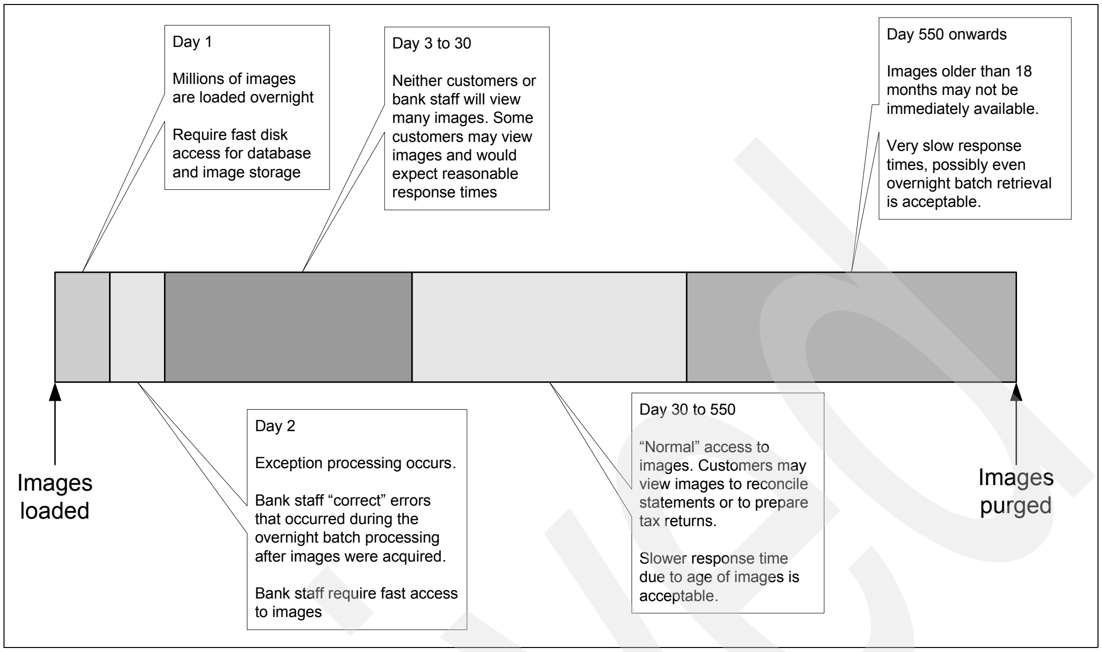


Figure 7-5 Storage lifecycle of images

Mapping the requirements onto the storage

In a CMOD solution the database would almost always be placed on high speed protected storage (for example, SAN or NAS). The disk system used for the database, the operating system and the database itself has to be tuned to operate at maximum efficiency.

The storage for the check images themselves will depend upon the age of the image in the archive and they must be seamlessly migrated between storage locations as the storage policy dictates.

The cache areas where, most commonly, all newly loaded objects are stored, is usually on fast protected local, SAN or NAS attached disk.

For longer term storage, we would look at providing more cost effective storage locations for the images. Possible storage locations could include NAS, tape or optical controlled via Tivoli Storage Manager.

Therefore, for the check image system, one possible solution for the storage for the system could be as follows:

1. SAN or NAS — Database — used for life of the images
2. SAN or NAS — CMOD cache storage — used for 30 days
3. SAN or NAS — First Tivoli Storage Manager storage location — used for 180 days
4. NAS — Second Tivoli Storage Manager storage location — used from day 180 to day 550
5. Optical or tape — Third storage location — - used from day 550 to end of life

7.2.6 Summary

From the previous sections, we can see that Content Manager and Content Manager OnDemand have the capabilities to leverage the IBM Information Infrastructure to protect your data and ensure that data:

- Remains secure
- Has its integrity protected and assured
- Is retained according to corporate regulations and in a cost effective way
- Remains available for retrieval when access to data is required

Part 3

Case studies

In this part of the book, we illustrate various real-world implementations of the IBM ECM suite of products and show how they exploit IBM storage products.

8

Case Study I: Online Banking Services

According to PEW Internet research, from 2002 to 2004, the number of users banking over the Internet grew 53%[1]. This case study assumes that a fictitious company, the ITSO Banking Corporation, is attempting to capitalize on this trend by offering an end-to-end SOA solution to small banks wanting to offer online banking services.

ITSO's application will offer services such as online bill payments, deposits, and withdrawals to the customers of banks they partner with. By using its volume purchasing power, ITSO will procure assets at a discount and pass along a portion of the savings to their partners, who will as a result be better positioned to compete with larger banks. In consideration of this service, ITSO Banking Corporation will collect a fee for every transaction.

Banks having an existing online service will be able to deliver a stronger product at a significant cost savings. Banks without these services will benefit from a proven solution and be able to bring an online banking product to market sooner than they could implement themselves. In short, ITSO will be able to deliver better service at a lower total cost of ownership. Although ITSO is well funded, maintaining cost control without sacrificing service will be a critical factor to establishing value with their customers.

[1] Susannah Fox. "The state of online banking," *PEW Internet and American Life Project,* February 2005: 1-3 <http://www.pewinternet.org/pdfs/PIP_Online_Banking_2005.pdf> 12 February 2008

8.1 Requirements and assumptions

ITSO is actively marketing its services and is promising four potential partners a superior online banking solution, brought to market quickly, and at a substantial cost savings compared to maintaining a stand alone application. The implementation of hardware and software will be based on a repeatable pattern, allowing ITSO to quickly deliver on the obligations of newly signed partnership agreements. The ability to quickly deliver a functional service to a prospective bank is critical to ITSO's value proposition and is key to closing new business.

The cost advantage of purchasing infrastructure in volume allows ITSO to offer a turn key solution at a cost savings. This coupled with the ability to draw on the expertise of multiple banking partners enables ITSO to develop a cogent and sound platform, thus providing a best-of-breed online banking experience.

ITSO Bank is head quartered in Los Angeles, California and has partnered with four banks for its product launch, one in Culver City near Los Angeles, another in Houston Texas, and two on the east coast in Boston Massachusetts and Columbus Georgia respectively. The banking habits of customers differ region to region, so each bank has provided an estimate of potential users of the online service and what they perceive their use case will entail. Surveys from each bank were taken and used to create a reasonable projection of how many users would be online and at what times and these numbers are used to help define the technical requirements of the project.

To deliver a consistent and reliable service, ITSO Banking Corporation requires an ECM environment capable of handling large volumes of data and accommodate seasonal transaction peaks without affecting performance. ITSO will be responsible for processing transactions from a number partner banks across the country and must provide continuous availability around the clock.

To establish value early, ITSO will guarantee the date that the banking application will be available to customers. Thus, the lead time required to start service must be less than the time the partner would have to implement a solution themselves. By ensuring that ITSO demonstrates value early, partners can quickly realize a return on investment and eliminate funding as a factor in preventing a partner from purchasing the service.

The services provided by the application will be available in a programatic fashion as well as though a user interface. Any capability demonstrated by a person using a user interface will be made available as part of an SOA framework. This allows the partner banks to enhance and extend their services and provide integration opportunities with their internal systems.

The ITSO ECM infrastructure must be able to accommodate both organic growth and planned expansions. This requirement is driven by the natural growth of the partner banks clientele and ITSO's pursuit of new business. As new partners are brought online, a large number of accounts will be added all at once, and these changes must occur without affecting the customers of other banking partners. The ITSO application will be available 24x7, which requires infrastructure additions to be completed during production hours, while the application online and in use.

8.1.1 Determining business requirements

Meetings with bank partners and ITSO executive staff have yielded a number of requirements. ITSO is funded by a private equity firm who has mandated strict cost guidelines to ensure expenses increase at a rate that will not outpace revenue. The combined requirements are outlined here:

- All infrastructure costs must be aggressively negotiated. The business model depends on procuring software, hardware, and services at a rates more favorable than its customers.
- Within ten business days of the close of a new contract, a new bank partner must be able utilize the basic functions of the service.
- No interruption can occur in the service at any time and no single banking partner is allowed to affect another partner.
- Infrastructure changes must not adversely affect the service in any way.
- Bank partners must be able to integrate their services with ITSO with the same capabilities available to their customers.
- Bank partners must be able to reconcile records and make manual changes to correct banking errors.
- Partner banks must be able to offer their customers the ability to fax in banking requests.
- The user interface intended for the customers of partner banks must be brandable. It must not be obvious that the partner has out sourced the service to a third party.
- Choices in infrastructure at inception must not preclude vendor choice in the future. The platform must be flexible enough to allow ITSO to switch between vendors and promote competition between existing vendors and new ones.
- The service must assert geographic redundancy.

8.1.2 Determining technical requirements

From the business requirements, technical requirements further define the obligations that the ECM solution must meet. Each business requirement is assigned to an equivalent technical requirement to ensure that the direction of the business is reflected in the capabilities required of the infrastructure:

- All infrastructure costs must be aggressively negotiated. The business model depends on procuring software, hardware, and services at a rate more competitive than its partners.
  - The initial environment will be built to support the application and the four partner banks though the first year. This initial capital expenditure will be substantial and must allow for volume discounting on all aspects of the environment.
- Within ten business days of the close of a new contract, a new bank partner must be able utilize the basic functions of the service.
  - The environment will be designed to scale horizontally using load balancer and application server clusters. Capacity to accommodate new partners will be turned up on demand using a easily repeatable provisioning process.
- No interruption can occur in the service at any time and no single banking partner is allowed to affect another partner.
  - Infrastructure will be initially sized 50% larger than the maximum projected load required and new systems will be added before utilization reaches 75%. This allows for a portion of the environment to be offline for maintenance and to accommodate organic growth. Customer data will be logically segregated based on the bank partner they do business with.

- Infrastructure changes must not adversely affect the service in any way.
  - Whenever possible, server infrastructure will be built in manner allowing any server node to fulfill the role of any other server. This will be achieved by booting servers from a SAN and organizing them into roles managed by virtual IP addresses. By moving servers between VIP groups, the environment can be rebalanced as necessary. This mass simplification allows for easy provisioning and adding of capacity on the fly. By removing a server from a VIP, and having extra capacity on hand, systems can be changed added and removed without affecting the application or its users.
- Bank partners must be able to integrate their services with ITSO with the same capabilities available to their customers.
  - The ECM application will make liberal use of Web services to provide an end-to-end SOA solution.
- Bank partners must be able to reconcile records and make manual changes to correct banking errors.
  - Bank partners must be provided with a Web interface suitable for use by a business analyst.
- Partner banks must be able to offer their customers the ability to fax in banking requests.
  - A service capable of receiving a fax must be able to interpret the contents of a form filled out by hand and execute the described financial transaction automatically.
- The user interface intended for the customers of partner banks must be brandable. It must not be obvious that the partner has out sourced the service to a third party.
  - An AJAX driven J2EE Web application will be provided as a baseline online interface. A style sheet has to control the overall look and feel of the interface and strategically placed banners must allow the partner to apply its brand. Further customization is welcomed, but will be the responsibility of the partner.
- Choices in infrastructure at inception must not preclude vendor choice in the future. The platform must be flexible enough to allow ITSO to switch between vendors and promote competition between existing vendors and new ones.
  - First preference will be given to open standards platforms and vendors with a commitment to high interoperability. Whenever possible, closed platforms will be avoided as will technologies which specifically tie the business to a single vendor.
- The service must assert geographic redundancy.
  - The ECM solution must be available at all times. This requires replicating all of the data to a remote data center.

8.2 Designing the solution

The ITSO application must be able to allow a bank's customers to perform basic banking tasks and the bank's financial analysts to be able reconcile and correct errors. To achieve this, the ITSO environment will be built on an SOA framework powering two Web interfaces. The first interface is intended for bank customers to login and do their banking online. It will be brandable by the partner banks and based on an entirely new J2EE application. The second interface is a customized version of Workplace intended for business analysts at partner banks to review transaction history and reconcile accounts against their own records. Both interfaces will provide services based on the content and workflow capabilities of IBM FileNet P8 platform.

To best address the interoperability requirements of the partner banks, every online banking operation will be expose Web service interfaces to the respective partner banks to allow programatic access to the online banking service. This will allow the bank a high degree of integration options and the freedom to integrate the solution with their own systems in the language they choose.

Web services will be created to provide common functions such as deposits, transfers, withdrawals, and some additional analyst functions, namely, account creation, suspension, deletion, and transaction reconciliation. A transaction history Web service will be available for partner banks to periodically update their records. Each customer is represented by a folder object containing information relevant to their account, and in that folder are a number of custom and document objects that will comprise the entirety of the account as illustrated by Figure 8-1.

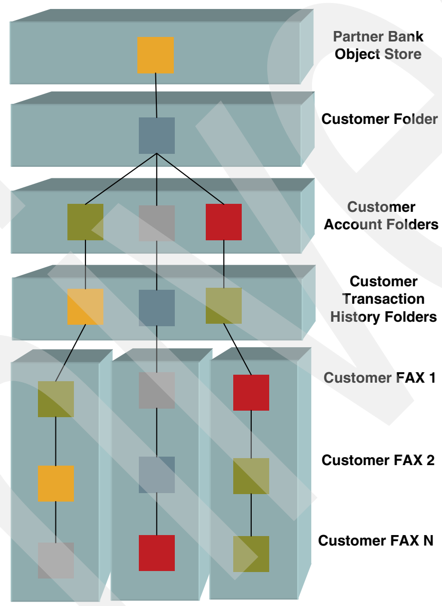


Figure 8-1 Content structure

Figure 8-2 is a logical breakdown of the ECM solution design as it would appear to each partner. The customer banking interface implements functions in the Customer SOA interface. This design ensures changes in the banking user interface do not interfere with the functions of the ECM application. This requirement allows each banking partner to tailor the look and feel of the online banking experience without interfering with the entire ECM environment. The Customer SOA interface is a set of Web services that handle user transactions. Users require the ability to deposit funds, transfer funds between accounts, pay bills, and request withdrawals. Each method controls a Process Engine Workflow that executes transactions, validations, and processes updates.

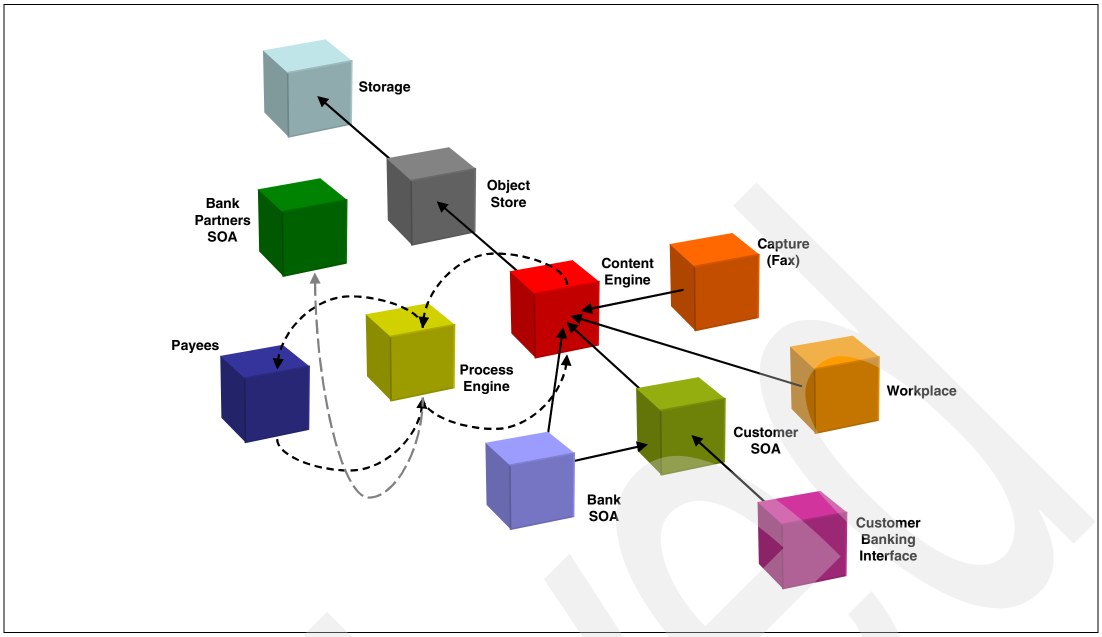


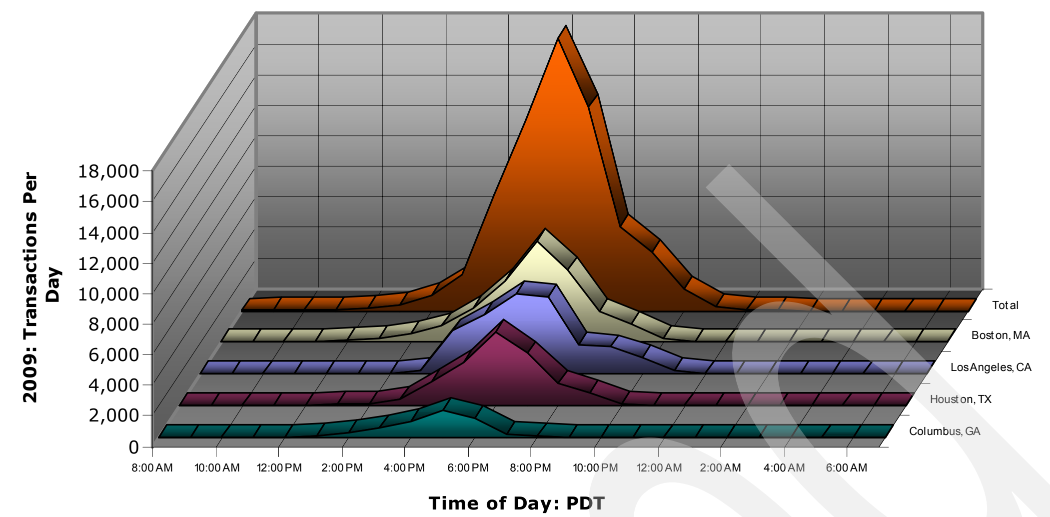
Figure 8-2 Logical ECM design

The Bank SOA block represents a set of Web services intended for use by the partner banks. These Web services play two roles. The first one is to allow business analysts to manually manage customer data and workflow using Workplace. The second one is to allow programatic access. Process Engine serves as the point of integration to process deposits, withdrawals, and bill payments by connecting to payee and bank partner's Web services. Finally, Capture will be used to process banking transactions that arrive as faxes. When a fax arrives, it is processed by Capture, which scans the fax image. Only faxes of bank supplied forms will be accepted. When a form arrives, the customer data is read using optical character recognition. The form is then imported as a document object and filed in the customer's folder.

8.2.1 Projecting volume growth and use patterns

ITSO Banking Corporation plans to launch with four banks servicing 2.65 million total users. Of this amount, only a portion of the account holders will be logged in and actively performing transactions. The launch product will handle transfers, withdrawals, deposits, and bill payments over the Internet as well as by fax.

Figure 8-3 illustrates the data projecting page view activity for this first year. It is assumed that the number of page views will be determined by a factor of the number of steps required to complete transaction multiplied by the average number of transactions per login, multiplied again by the number of users. Although this type of data is primarily used to size presentation tier components, it also gives an indication of how the storage will be consumed and what times of day the storage will be busiest.



UTC	Los Angeles, CA		Houston, TX		Boston, MA		Columbus, GA		Total Page Views / Day
4:00 PM	0	8:00 AM	0	10:00 AM	60	11:00 AM	0	11:00 AM	60
5:00 PM	0	9:00 AM	0	11:00 AM	128	12:00 PM	0	12:00 PM	128
6:00 PM	0	10:00 AM	8	12:00 PM	255	1:00 PM	0	1:00 PM	263
7:00 PM	0	11:00 AM	45	1:00 PM	518	2:00 PM	15	2:00 PM	578
8:00 PM	8	12:00 PM	248	2:00 PM	1,035	3:00 PM	83	3:00 PM	1,373
9:00 PM	45	1:00 PM	495	3:00 PM	2,070	4:00 PM	420	4:00 PM	3,030
10:00 PM	248	2:00 PM	990	4:00 PM	4,148	5:00 PM	2,123	5:00 PM	7,508
11:00 PM	1,740	3:00 PM	2,978	5:00 PM	8,303	6:00 PM	4,259	6:00 PM	17,279
12:00 AM	20,897	4:00 PM	11,925	6:00 PM	16,610	7:00 PM	7,666	7:00 PM	45,668
1:00 AM	29,387	5:00 PM	21,465	7:00 PM	29,898	8:00 PM	12,777	8:00 PM	73,051
2:00 AM	39,182	6:00 PM	35,775	8:00 PM	49,829	9:00 PM	9,086	9:00 PM	101,075
3:00 AM	37,876	7:00 PM	25,440	9:00 PM	35,434	10:00 PM	1,298	10:00 PM	86,533
4:00 AM	14,367	8:00 PM	11,130	10:00 PM	15,503	11:00 PM	255	11:00 PM	51,589
5:00 AM	13,061	9:00 PM	6,360	11:00 PM	8,859	12:00 AM	85	12:00 AM	57,779
6:00 AM	7,836	10:00 PM	1,058	12:00 AM	1,770	1:00 AM	28	1:00 AM	35,075
7:00 AM	1,560	11:00 PM	212	1:00 AM	590	2:00 AM	9	2:00 AM	13,289
8:00 AM	308	12:00 AM	42	2:00 AM	197	3:00 AM	3	3:00 AM	6,867
9:00 AM	60	1:00 AM	8	3:00 AM	66	4:00 AM	1	4:00 AM	1,184
10:00 AM	8	2:00 AM	0	4:00 AM	22	5:00 AM	0	5:00 AM	241
11:00 AM	0	3:00 AM	0	5:00 AM	7	6:00 AM	0	6:00 AM	50
12:00 PM	0	4:00 AM	0	6:00 AM	0	7:00 AM	0	7:00 AM	0
1:00 PM	0	5:00 AM	0	7:00 AM	0	8:00 AM	0	8:00 AM	0
2:00 PM	0	6:00 AM	0	8:00 AM	0	9:00 AM	0	9:00 AM	0
3:00 PM	0	7:00 AM	0	9:00 AM	0	10:00 AM	0	10:00 AM	0

Figure 8-3 First Year Page Views

The section highlighted in the middle of the spreadsheet indicates the highest periods of activity and also communicates how the time zone differences affect the load characteristics. The sections highlighted on the far left and far right indicate the aggregate of the activity in their middle and allows the environment to be more accurately sized because it is a reflection of all the activity. According to this projection, the peak hours of activity for this application will be between 1:00PM and 11:00PM Pacific time. This will be the period where the most load will be placed on the application. As such, the application will be sized to accommodate the peak traffic spikes expected during this time frame.

Figure 8-4 is spreadsheet projecting storage growth for the first year and is based on of transaction volumes. By aggregating the data provided by the partner banks, ITSO has calculated that the adoption of transaction over fax will be a modest percentage of the total population and increase in following years as users become more comfortable with the service. Ensuring that the storage is expanded at a rate commensurate with growth will require ITSO to not only monitor the adoption of online users, but also how many of those users also use fax. The growth of the ITSO partner portfolio will also cause these numbers to grow.

UTC Time	Los Angeles, CA		Houston, TX		Boston, MA		Columbus, GA		Total Fax Storage / Day in MB	Total Fax Storage / Year in GB
4:00 PM	0	8:00 AM	0	10:00 AM	1	11:00 AM	0	11:00 AM	1	0
5:00 PM	0	9:00 AM	0	11:00 AM	3	12:00 PM	0	12:00 PM	3	1
6:00 PM	0	10:00 AM	0	12:00 PM	5	1:00 PM	0	1:00 PM	5	2
7:00 PM	0	11:00 AM	1	1:00 PM	10	2:00 PM	0	2:00 PM	12	4
8:00 PM	0	12:00 PM	5	2:00 PM	21	3:00 PM	2	3:00 PM	27	10
9:00 PM	1	1:00 PM	10	3:00 PM	41	4:00 PM	8	4:00 PM	61	22
10:00 PM	5	2:00 PM	20	4:00 PM	83	5:00 PM	42	5:00 PM	150	54
11:00 PM	35	3:00 PM	60	5:00 PM	166	6:00 PM	85	6:00 PM	346	123
12:00 AM	418	4:00 PM	239	6:00 PM	332	7:00 PM	153	7:00 PM	1,142	407
1:00 AM	588	5:00 PM	429	7:00 PM	598	8:00 PM	256	8:00 PM	1,871	667
2:00 AM	784	6:00 PM	716	8:00 PM	997	9:00 PM	182	9:00 PM	2,677	954
3:00 AM	758	7:00 PM	509	9:00 PM	709	10:00 PM	26	10:00 PM	2,001	713
4:00 AM	287	8:00 PM	223	10:00 PM	310	11:00 PM	5	11:00 PM	825	294
5:00 AM	261	9:00 PM	127	11:00 PM	177	12:00 AM	2	12:00 AM	567	202
6:00 AM	157	10:00 PM	21	12:00 AM	35	1:00 AM	1	1:00 AM	214	76
7:00 AM	31	11:00 PM	4	1:00 AM	12	2:00 AM	0	2:00 AM	47	17
8:00 AM	6	12:00 AM	1	2:00 AM	4	3:00 AM	0	3:00 AM	11	4
9:00 AM	1	1:00 AM	0	3:00 AM	1	4:00 AM	0	4:00 AM	3	1
10:00 AM	0	2:00 AM	0	4:00 AM	0	5:00 AM	0	5:00 AM	1	0
11:00 AM	0	3:00 AM	0	5:00 AM	0	6:00 AM	0	6:00 AM	0	0
12:00 PM	0	4:00 AM	0	6:00 AM	0	7:00 AM	0	7:00 AM	0	0
1:00 PM	0	5:00 AM	0	7:00 AM	0	8:00 AM	0	8:00 AM	0	0
2:00 PM	0	6:00 AM	0	8:00 AM	0	9:00 AM	0	9:00 AM	0	0
3:00 PM	0	7:00 AM	0	9:00 AM	0	10:00 AM	0	10:00 AM	0	0
Totals	**3,332**		**2,364**		**3,506**		**762**		**9,963**	**3,551**

Figure 8-4 First year content storage growth

The projection above indicates that in the first year, 3.5 TB of content will arrive via fax, but the model does not reflect growth in each bank's customer base nor the addition of additional banking partners. The omission of these factors is purposeful. ITSO plans to build a storage platform with the ability to continuously adapt as the consumption statistics change. In doing so, ITSO does not have to purchase disk years in advance, but instead, storage can be purchased in smaller increments proportional to the growth of the business. By doing so, ITSO will be able to take advantage of the declining cost of storage.

Figure 8-5 is similar to Figure 8-4 but instead focuses on projected database growth. The growth of the content will only be affected by faxes, but metadata growth will be affected by all online activity. The content model in Figure 8-1 on page 157 indicates each user as having a minimum of three objects, with a new object being created with each transaction. The history of these objects is recorded using the transaction folder object. These statistics were used to generate first year projections. As with content storage, database storage must be able to grow as users are added. Additionally, however, the growth rate will be affected by how often a customer uses online banking over time. For example, a customer unfamiliar with computers or online banking might only use the service to check their account balance. As this user becomes more familiar with the interface, they might decide to use it to pay bills. This change in behavior affects transactions, therefore page views, and eventually storage.

UTC Time	Los Angeles, CA		Houston, TX		Boston, MA		Columbus, GA		Total Database Storage / Day in MB	Total Database Storage / Year in GB
4:00 PM	0	8:00 AM	0	10:00 AM	0	11:00 AM	0	11:00 AM	0	0
5:00 PM	0	9:00 AM	0	11:00 AM	0	12:00 PM	0	12:00 PM	0	0
6:00 PM	0	10:00 AM	0	12:00 PM	0	1:00 PM	0	1:00 PM	0	0
7:00 PM	0	11:00 AM	0	1:00 PM	1	2:00 PM	0	2:00 PM	1	0
8:00 PM	0	12:00 PM	0	2:00 PM	2	3:00 PM	0	3:00 PM	2	1
9:00 PM	0	1:00 PM	1	3:00 PM	4	4:00 PM	1	4:00 PM	5	2
10:00 PM	0	2:00 PM	2	4:00 PM	7	5:00 PM	4	5:00 PM	13	5
11:00 PM	3	3:00 PM	5	5:00 PM	14	6:00 PM	7	6:00 PM	29	10
12:00 AM	35	4:00 PM	20	6:00 PM	28	7:00 PM	13	7:00 PM	97	34
1:00 AM	50	5:00 PM	36	7:00 PM	51	8:00 PM	22	8:00 PM	158	56
2:00 AM	66	6:00 PM	61	8:00 PM	84	9:00 PM	15	9:00 PM	227	81
3:00 AM	64	7:00 PM	43	9:00 PM	60	10:00 PM	2	10:00 PM	169	60
4:00 AM	24	8:00 PM	19	10:00 PM	26	11:00 PM	0	11:00 PM	70	25
5:00 AM	22	9:00 PM	11	11:00 PM	15	12:00 AM	0	12:00 AM	48	17
6:00 AM	13	10:00 PM	2	12:00 AM	3	1:00 AM	0	1:00 AM	18	6
7:00 AM	3	11:00 PM	0	1:00 AM	1	2:00 AM	0	2:00 AM	4	1
8:00 AM	1	12:00 AM	0	2:00 AM	0	3:00 AM	0	3:00 AM	1	0
9:00 AM	0	1:00 AM	0	3:00 AM	0	4:00 AM	0	4:00 AM	0	0
10:00 AM	0	2:00 AM	0	4:00 AM	0	5:00 AM	0	5:00 AM	0	0
11:00 AM	0	3:00 AM	0	5:00 AM	0	6:00 AM	0	6:00 AM	0	0
12:00 PM	0	4:00 AM	0	6:00 AM	0	7:00 AM	0	7:00 AM	0	0
1:00 PM	0	5:00 AM	0	7:00 AM	0	8:00 AM	0	8:00 AM	0	0
2:00 PM	0	6:00 AM	0	8:00 AM	0	9:00 AM	0	9:00 AM	0	0
3:00 PM	0	7:00 AM	0	9:00 AM	0	10:00 AM	0	10:00 AM	0	0
Day Total	**282**		**200**		**297**		**65**		**843**	**301**
1st Year GB	**102**		**73**		**107**		**24**			

Figure 8-5   First Year Database Storage Growth

The absence historical data makes it difficult to accurately project how business growth and customer behavior will affect storage volumes. An educated assumption of how the application is intended to function will instead be used to define a model from which a reasonable projection can be made. After the application has been in production, historical statistics will be used to model application behavior.

Figure 8-6 illustrates the projected use pattern for fax content. There is little benefit to having deposit and withdrawal capabilities over fax because there is no way to transfer a monetary instrument between the fax originator and the bank. Limiting the scope of services to payments and transfers allows us to carefully project how these transactions might be handled over a fifteen day period.

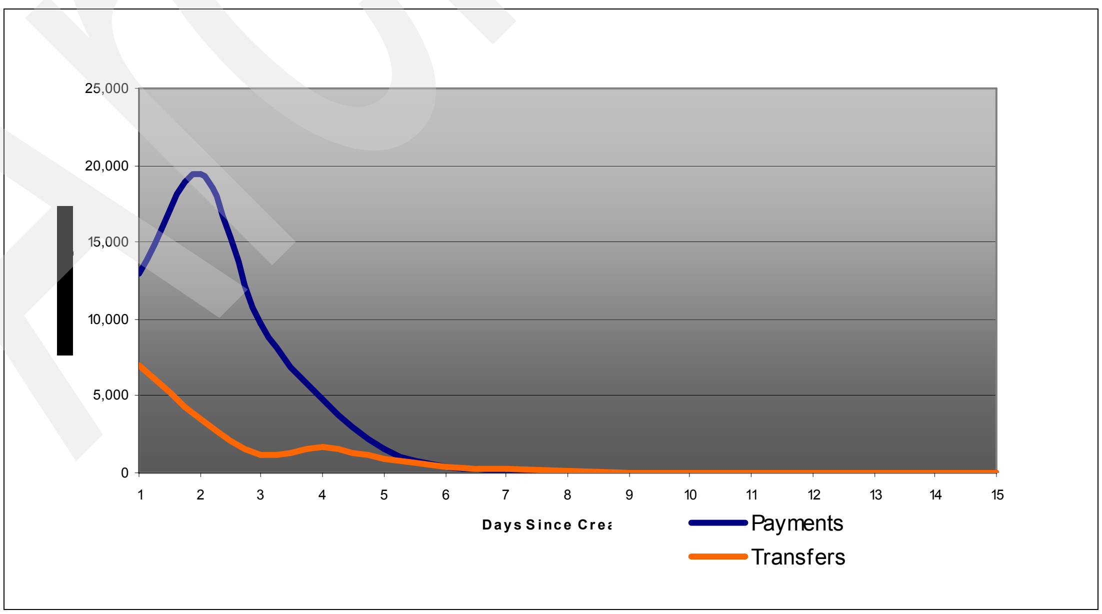


Figure 8-6   Content Use Decay Model

To create a model on which to base projections, an outline of how a partner affects the storage in the context of a single day must be created. The single day model must then be tracked until all of the work in that day is completed. As faxes arrive, they are queued for processing and some of these transactions will be orders to transfer funds while others will be for bill payments. A workflow which co-ordinates the processing of the order causes funds to be transferred to a payee or to another account specified on the form. For the form to be interpreted by the application, it must first undergo an OCR process that extracts the text from the form. Part of this process includes a verification by a bank employee who physically inspects the form to ensure that the data was correctly captured. As partner banks begin processing, a sharp spike is expected in database growth, which reflects the new objects that are created with each transaction. Over time, the rate of object creation falls to reflect the fact that not all faxes are guaranteed to be processed the same day, and that some might be held up to call back the customer to verify the form.

A similar effect is observed for money transfers, but in smaller volumes. Reflected in the graph is the anticipation that transfers can require less visual verification, and that at around day four, drafts that were held cause a small increase in request activity as they clear before trailing off again. What this tells us about the storage, is the relative value of the faxes as they age. A fax is not expected to be accessed after seven days. All of the activity occurs in the first few days when the fax arrives. As it gets older, its value trails off because there are no requests to view it. This creates a problem from a storage value standpoint. When the fax is new, high speed storage is required to ensure that the fax images are retrievable in a time frame suitable for the Web. After seven days, the content no longer requires the speed of the storage because it is accessed infrequently, but it still cannot be purged for records retention purposes. The cost of maintaining low value data on high cost storage is less than ideal.

The following technical requirements will be expanded from section 8.1.2, “Determining technical requirements”:

- Infrastructure changes must not adversely affect the service in any way.
  - Whenever possible, server infrastructure will be built in manner allowing any server node to fulfill the role of any other server. This will be achieved by booting servers from a SAN and organizing them into roles managed by virtual IP addresses. By moving servers between VIP groups, the environment can be rebalanced as necessary. This mass simplification allows for easy provisioning and adding of capacity on the fly. By removing a server from a VIP, and having extra capacity on hand, systems can be changed added and removed without affecting the application or its users.
    - Content must reside on tiered storage. Data in the application has a natural lifespan where the value of the data decreases over time. When the data reaches a predefined age, it must be moved from high cost storage to low cost storage.
    - Archived content can be audited in the future. During an audit, content in question must be moved to a temporary storage area to ensure timely processing.
- The service must assert geographic redundancy.
  - The ECM solution must be available at all times. This requires replicating the content and metadata to a remote data center.
    - SAN storage must be replicated, but to accommodate differing access patterns, the data might not saved to a single storage device. Additionally, as the business grows, multiple storage arrays might be dedicated to different types of data. ITSO is legally liable for every transaction, so the replicated copy must be in a transactional valid state, regardless of where object data is resides.
- No interruption can occur in the service at any time and no single banking partner is allowed to affect another partner.

– Infrastructure will be initially sized 50% larger than the maximum projected load required and new systems are added before utilization reaches 75%. This allows for a portion of the environment to be offline for maintenance and to accommodate organic growth. Customer data will be logically segregated based on the bank partner they do business with.
  - SAN Storage for metadata will be sized 50% larger than the projected consumption for the year and will be expanded before capacity reaches 75%.
  - SAN Storage for the database will be sized 50% larger than the projected consumption for the year and will be expanded before capacity reaches 75%.
  - SAN Storage for a Content Engine Cache will be used to improve performance of transitive activity relating to content. The content cache allows content which is active to be cached on high speed storage, but archived to lower cost storage for long term archival.

8.3 Overall solution architecture

ITSO Banking Corporation, after finalizing its business requirements, has derived its ECM functional requirements. The functional requirements were based on a study of data volumes and use patterns expected. Technical requirements have been made to reflect the output of the study. The remaining task is to develop the infrastructure design and ensure that it meets or exceeds the capabilities required by the business. The design is broken down into two sections, software and hardware.

8.3.1 Software

Next we describe the P8 software components used to build the solution.

IBM FileNet Content Engine 4.0

Content Engine will directly manage the content and metadata for the ITSO application. Each partner bank will be assigned an object store which logically segregates each banking environment from the other. This allows each bank to introduce changes to their online banking application without interfering with other banks and their customers. Although mass changes would be discouraged, no attempt would be made to restrict the capabilities of the platform.

Content Engine instances will be co-located with Process Engine and each Content Engine instance will have a Process Engine connection point configured to use the loopback address of the server. This configuration negates requiring a managed VIP for Process Engine and reduces the amount on wire network traffic between these components. Running multiple application components improves server utilization and thus is a better value from both recurring and capitalized cost perspectives.

IBM FileNet Process Engine 4.0

Process Engine will manage all of the interactions between partner connections, payment services, and provide the banking transaction services used by the proposed Web interfaces. Each bank will use a dedicated isolated region allowing the process related functionality for each bank to be separate and distinct. The process isolation also enables different partner banks to connect to distinct bill payment services allowing them to retain their established partnerships.

As previously stated, Process Engine will be co-located with Content Engine and connection points for each of the isolated regions will connect over the loop back address. Having these components on a single machine allows us both to better utilize the hardware and reduces the complexity, which helps reduce the time to recovery if a component fails.

IBM FileNet Process Analyzer 4.0

Process Analyzer will play two roles. First as an ad-hoc tool for reporting on historic activity for internal processes allowing partner banks to review workflow history and optimize their internal processes. Process Analyzer will also be used for billing purposes. Reports will be configured to allow ITSO to review past transaction history and bill their partners accordingly.

IBM FIleNet Application Engine 4.0

Application Engine will provide the content and process APIs as well as the Workplace features required for users to interact with the application. Custom banking interfaces will run on the same server as Application Engine. Application Engine is scaled horizontally using an application server cluster.

IBM FileNet System Monitor 4.0

FileNet System Monitor will be used to monitor activity from a content and process perspective in real time. FSM provides real time graphing capability to allow application administrators to monitor FileNet applications as they are in use. This will be a valuable performance management tool for ITSO administrators. Additionally, this tool will allow application administrators to meet the requirement of managing capacity. FSM allows administrators to accurately correlate content related statistics back to the application behaviors and server performance statistics. Daily use of FSM will help determine when upgrades have to be made.

FileNet System monitor will share a machine with FileNet Enterprise Manager and an instance of WebSphere dedicated to managing application server clusters.

IBM FileNet Capture Professional 5.1

Capture will be responsible for processing inbound faxes. When forms are faxed, the document image is scanned using optical character recognition. After a quality check of captured metadata, the content is release to Content Engine, and a workflow is launched to process the request. When a fax cannot be properly scanned, it is routed to a business analyst who will make corrections and launch the appropriate workflow.

IBM DB2 Universal Database 9.1

The ECM application components will use DB2 as its database with each bank having a dedicated instance for an object store. A shared instance will be dedicated to the Content Engine GCD (Global Configuration Data). Another instance is shared between all partners for the Process Engine, because Process Engine is data is segregated at the application level, negating the benefit of having dedicated instances.

Unlike other ECM components, DB2 will run on dedicated hardware better suited for vertical scalability. As partner banks are signed to ITSO's service, an instance of DB2 is created to support the partner's metadata. DB2 will made highly available using HACMP™. Continuous availability will be provided during upgrades using the modular SMP capability of the p570, a feature that allows for hardware upgrades while the operating system is running.

IBM DB2 provides the capability of creating of database instances in a scripted manner, which allows administrators to automate the configuration changes. This helps ITSO to meet the tight provisioning guidelines specified by their business requirements.

IBM WebSphere Application Server Network Deployment 6.0

IBM WebSphere Application Server Network Deployment (WAS ND) is a multi-platform J2EE application server combined with centralized management, caching, and load balancing capabilities. Content Engine and Application Engine run in an application server, and WebSphere Network Deployment will allow ITSO administrators to manage the ECM environment as a whole. Thus, ITSO can run the environment with a better server-to-administrator ratio than what might be typically found in the industry. WAS ND dramatically reduces the configuration overhead of adding object stores and of managing the overall number of application servers.

WAS ND also ships with a robust load balancing capability at the edge, for improving user response time with the user interface, and at the content layer for improving EJB performance. Both load balancing services provided by WebSphere clustering provide seamless fail over capability, a key business requirement for ITSO. A load balancer will be used to provide availability in the IBM HTTP server.

The WebSphere Application Server deployment manager will run on a machine dedicated to administrative tools. FileNet System Monitor and FileNet Enterprise Manager will also be installed here. This allows the application environment to be centrally managed and segregates these systems from application traffic that might otherwise interfere with the management of the ECM solution.

As a new partner banks are signed on, they will require the WebSphere Application Server environment to be configured accommodate a new object store. Each object store requires two JNDI references to connect to the database. The Jython scripting features of WebSphere allow administrators to automate the configuration changes, which helps ITSO to meet the tight provisioning guidelines specified by their business requirements.

8.3.2 Hardware

The following is a manifest of the hardware components used to build the solution.

Server infrastructure

Figure 8-7 illustrates how the servers were matched to the performance characteristics outlined in 8.2.1, "Projecting volume growth and use patterns". The majority of the servers in the ECM solution will be running on IBM 560Q servers. The operating system and software for these servers will be run from SAN disk to allow any one server to assume the role of another. For example, if a server running Content Engine and Process Engine has a hardware failure, a server that would normally run Application Engine can be borrowed and booted using the LUN of the server that failed.

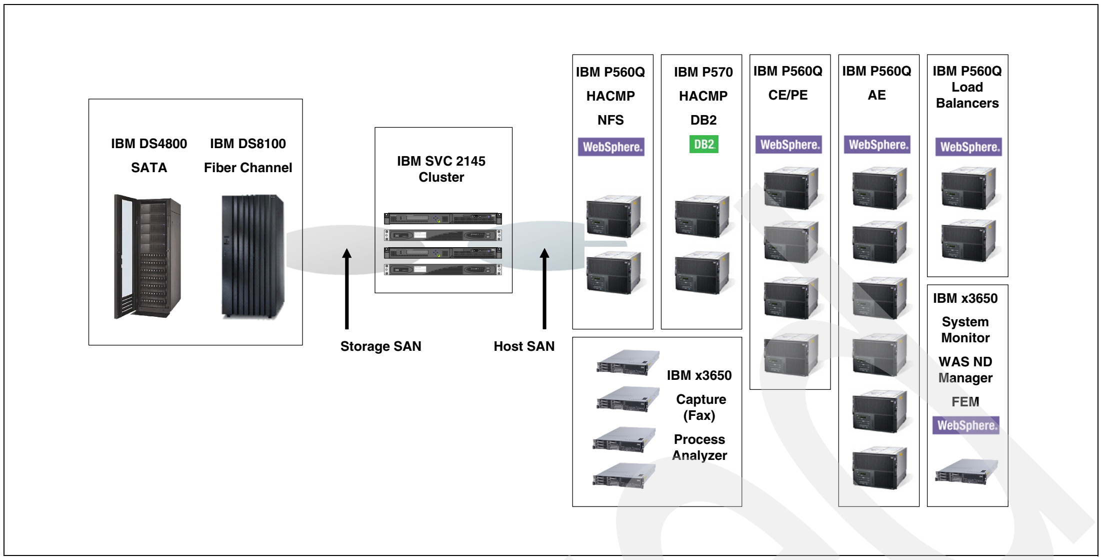


Figure 8-7 ITSO server infrastructure

This architecture allows for better flexibility, dramatically simplifies server administration, and improves the mean time to recovery of hardware failures. This also simplifies disaster recovery because exact copies of the operating system and software can be replicated to a remote location. The replication of patches and application maintenance is automatically available at the disaster recovery site, easing the burden of maintaining the disaster recovery site.

Storage infrastructure

FileNet ECM applications store data in separate locations, with each location being optimally tuned to the nature of how the data is accessed and its relative value to the business. In the case of ITSO, data is stored in the following manner:

- Content:
  - ITSO in its first year will store 3.5 terabytes of fax images. In seven years, assuming eight percent year over year growth, this will balloon to 284 terabytes to support only four partners. If ITSO wins a modest four additional partners a year (assuming a similar size), this number skyrockets to 1.2 petabytes. The cost to store this data and the infrequency it is accessed requires the use of lower cost SATA storage.
- Databases:
  - ITSO will be maintaining a database for every partner they sign as each object store requires an independent instance. At the end of year one, this will consume a little more 300 gigabytes and at the end of seven years, using the growth rate as the content, databases will consume 94 terabytes. To ensure optimal throughput the database will be stored on Fibre Channel disk. Several other databases must be also be maintained for Process Engine and Process Analyzer, the content of both have relational significance to data in the object store databases. Although the Process Engine will require high thoughput for transaction processing, the Process Analyzer will not, lending itself to be stored on more cost effective disk.

- Content Cache:
  - Directly conflicting with the requirements for content storage is the requirement to access content quickly when it is initially uploaded. As illustrated by the graph in Figure 8-6, the fax content loses most of its business value in the first seven days. Afterwards the content is kept for record keeping purposes. Large SATA volumes do not lend themselves well to high thoughput concurrent and random IO. A content cache on Fibre Channel disk provides better performance when it is required, without the expense of storing all of the content there. Objects can be seamlessly moved as users access older fax data.

ITSO's storage architecture leverages storage technologies best suited for the type of data and its access characteristics. The data contained in these silos requires each to be referentially accurate, meaning that a pointer in one application must refer to data in another application, but each storage technology replicates its own data independently, allowing the whole of the data to become referentially inaccurate. Out of sync records between systems could pose significant legal liabilities, given ITSO's business model. Referential integrity is especially important when workflows or an external systems of record are used. In these cases, referential accuracy requirements of the data extend beyond P8 application components. The aggregate of the information represents the actual business record, so for replicated copies or snapshots to be viable, these relationships must be somehow maintained.

Figure 8-8 illustrates how IBM SAN Volume Controller solves this problem for ITSO Banking. SVC is an in-band volume management appliance that connects to multiple LUNS in different arrays and presents them as aggregates. SAN Volume Controller physically intercepts all of the application writes, thereby providing a synchronization point. SVC controls and maintains the write order across all the applications regardless of which array stores the data or which host requests the write. Unifying multiple storage targets as virtual LUNS allows volumes to be resized and moved around as business requirements dictate. ITSO in turn is free to make changes to the storage architecture without impacting the application or forcing costly migrations. SVC is a standards based SAN appliance that is compatible with most storage vendors and operating systems.

Along the top of Figure 8-8 are the names of the partner banks ITSO intends to have signed for go-live. For simplicity the names are indicated by their geographic locations. The blocks directly beneath the bank names represent the areas in which data is stored. At the top, blocks indicate database storage that is configured for optimal performance and availability on large RAID 5 volumes mirrored across two IBM DS8100 disk arrays. The same type of storage is configured for the content cache area.

At the bottom is the content storage area which is configured for optimum space in an unmirrored RAID 5 arrangement. The content, however, is saved to SATA disk on an IBM DS4800. Each vertical column of blocks represents an SVC consistency group. A consistency group can be replicated or have a snapshot taken, which preserves the write order of the group. This will allow ITSO to intelligently restore the ECM application in a transactional valid state.

IBM SAN Volume controller provides the capability to move a LUN from one type of storage to another without shutting down the applications using the storage. This feature will allow ITSO to retire object stores to the lesser expensive disk on the DS4800. Just as content exhibits a diminishing value over time, so will old bank transactions. ITSO will have the freedom to create new object stores each year and move the previous year's object store to slower SATA storage.

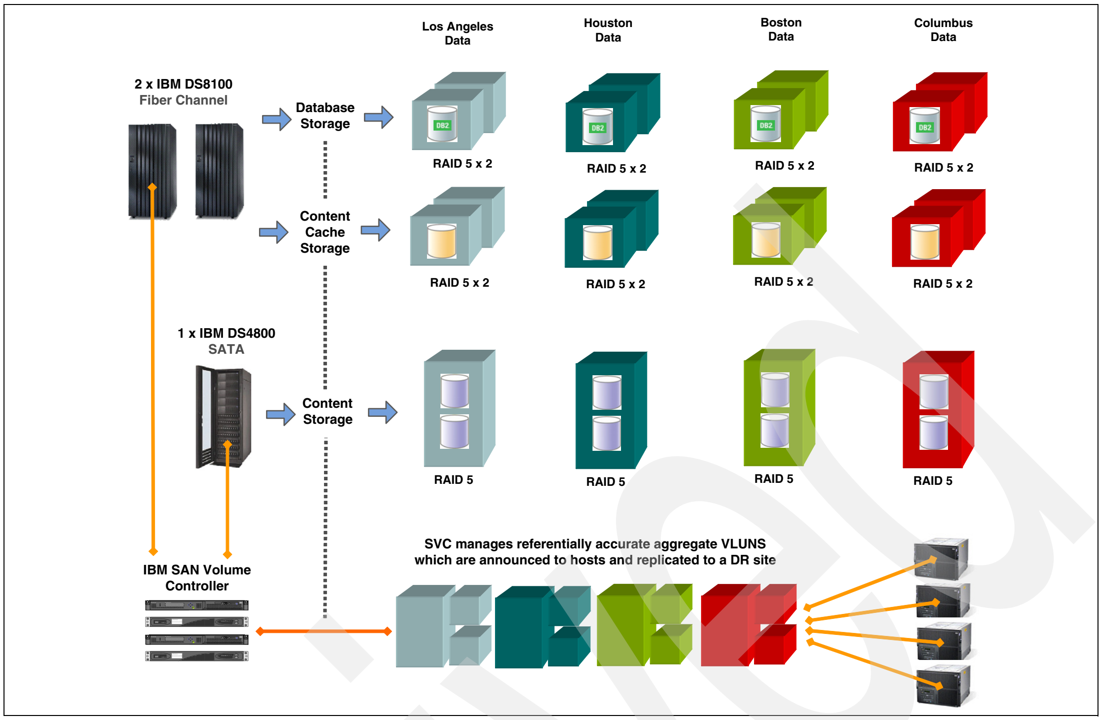


Figure 8-8 ITSO Storage Infrastructure

8.3.3 Solution summary

ITSO Banking Corporation, using this solution, can build a service that enables its customers to better compete with large banks without having to make an up-front investment in infrastructure. To hedge the cost of ITSO's own operational and capital expenses, this solution allows ITSO to deliver services such as online bill payments, deposits, and withdrawals with unfettered availability and create a significant cost advantage to its partners. In short, ITSO's low cost, high performance, highly available architecture allows its partners to enjoy a competitive advantage that they could not produce alone. The use of IBM storage allows ITSO to be flexible in implementation without restricting the storage infrastructure to a single vendor.

The IBM Information Infrastructure met ITSO's business requirements as outlined:

- All infrastructure costs must be aggressively negotiated. The business model depends on procuring software, hardware, and services at a rate more competitive than its partners.
  - IBM Web Sphere, IBM DB2, and IBM SAN Volume controller allow ITSO freedom of server and storage infrastructure, allowing them to aggressively negotiate the price of recurring capital expenditures. IBM P8's Service Oriented Architecture allows partner banks freedom to choose vendors for bill payment requirements.
- Within ten business days of the close of a new contract, a new bank partner must be able utilize the basic functions of the service.
  - IBM WebSphere, IBM DB2, and IBM FileNet P8 allow new bank partners to adopt a cradle-to-grave banking application in an automated manner, reducing the time required for a partner to realize a return on investment, and all while lowering the cost of administration.

- No interruption can occur in the service at any time and no single banking partner is allowed to affect another partner.
  - IBM FileNet P8 with IBM Storage provides application level segmentation of partner data, while IBM HACMP and IBM WebSphere ND provide continuous availability.
- Infrastructure changes must not adversely affect the service in any way.
  - IBM FileNet P8 and IBM SAN Volume Controller allow the ITSO application to be managed in the background independent of the code running the front end interfaces or the Web services used by outside entities.
- Bank partners must be able to integrate their services with ITSO with the same capabilities available to their customers.
  - IBM FileNet P8 allows any workflow to integrate with and expose Web services for a cradle-to-grave SOA.
- Bank partners must be able to reconcile records and make manual changes to correct banking errors.
  - IBM FielNet P8 provides a security model for back end management of the ITSO application allowing partner analysts to retrieve data and reconcile customer transactions.
- Partner banks must be able to offer their customers the ability to fax in banking requests
  - IBM FileNet P8, IBM FileNet Capture and IBM Storage provide the capability to fax in banking orders and have them processed automatically.
- The user interface intended for the customers of partner banks must be brandable. It must not be obvious that the partner has out sourced the service to a third party.
  - IBM FileNet P8 ships with an API allowing broad flexibility to create custom Web interfaces.
- Choices in infrastructure at inception must not preclude vendor choice in the future. The platform must be flexible enough to allow ITSO to switch between vendors and promote competition between existing vendors and new ones.
  - IBM WebSphere, IBM DB2, and IBM SAN Volume controller allow ITSO freedom of server and storage infrastructure freeing them from vendor lock in. IBM P8's Service Oriented Architecture and Component Integrator allows ITSO the freedom to integrate with whomever they choose.
- The service must assert geographic redundancy.
  - IBM San Volume Controller allows any LUN announced by SVC to be replicated while maintaining the write order fidelity of the entire application regards of which server it runs on, or the storage that the data is saved to.

9

Case Study II: Report Distribution and Data Archiving

In this case study, we discuss common storage requirements associated with archiving and distribution of reports generated by an SAP system, as well as archiving of inactive SAP data. This case is based on a number of customer implementations.

9.1 Introduction

Companies are facing increasing demands to make information widely available across their enterprises. Information from solutions that were originally created to support a single line of business now have to be tapped by other departments as well. In most cases the reuse and repurposing of information provides clear business advantages to companies.

The type of information that has to be available across the enterprise is not confined to metadata (database content), but also content (for example, documents) as well. ECM systems play a vital role in enabling this type of capability.

Companies are also increasingly aware of the type of information that are stored in their systems, and the actual utilization of that information by applications and/or end users. Many systems store vasts amounts of information that are rarely or never accessed; yet those systems are backed up and maintained because they are part of the enterprise. This results in storage and administrative expenses applied to information that is not required on a regular basis.

Thus, it becomes important to be able to distinguish between *active* and *inactive* information. Active information in this case can be defined as that which is likely to be accessed on a regular basis. Inactive, by the same token, is that information that is not likely to be accessed on a regular basis. However, the inactive information still has to be available for ad-hoc access and for compliance reasons.

In addition to being able to classify information as active and inactive, companies must have the capability to move the inactive data from primary production systems to other types of storage, while still maintaining proper access to the information. ECM systems also provide much of these capabilities, as described in this section.

9.2 Business scenario and requirements

This case describes a transportation equipment manufacturer in the United States, which sells its products through more than 3,000 dealerships across North America. The company interacts with both dealer representatives and end customers who purchases its products. Dealerships place their orders online or via phone, fax or e-mail. The company maintains a staff of about 30 dealer representatives to support that process. Dealer representatives try and ensure that all orders, order changes, payment, and credit adjustments are processed accurately and in a timely manner, and that the dealers are kept informed. They are also available to respond to any questions the dealers might have regarding specific accounts payable issues.

The company had some specific business objectives related to the accounts receivable process:

- **Improve accounts receivable process and dealer satisfaction:** Given that dealership orders constitute the revenue generating activity for the company, all activities supporting the order process have high visibility and high priority at all levels in the company. As such, the proper functioning and continuous improvement to the accounts receivable process is a top priority for the company. To improve on the process, and to increase dealer satisfaction, the company decided to allow dealers online access to their monthly statements.

- **Address growth of SAP database:** The order placement and fulfilment is done by the company's SAP system. Due to a significant increase in sales volume over the past years, the SAP database grew larger than initial projections. This in turn caused longer SAP database backup times, larger consumption of administrative and storage resources, and a decrease in system response time noticed by the SAP users.
- **Improve record retention process:** An additional concern is related to records maintenance. A duplicate print of all orders sent to customers was kept on file, in a traditional paper file archive, for a period of 7 years. When customers required reprints, the order had to be retrieved from the paper file and faxed or mailed to the customer. This manual process became increasingly slow and expensive as the volume of business grew. A work-around solution was reached by keeping the order print spools on disk for several months in the production system, allowing for reprints when necessary. This helped avoid the delay in the process since most reprints were for orders from the current or past month, which were present in the print spools. However, as print spools are large files, this caused additional usage of expensive storage in the SAP production system.

9.3 Accounts receivable process before ECM

Before implementation of ECM, the existing accounts receivable process consisted essentially of the creation and distribution of monthly statements created by the SAP system, plus an accounts receivable (AR) application that tracked change requests. Statements that reflect all orders and payments made in each month are created at the end of month by SAP, and sent via a print spool to the company print center. There, the statements are printed and mailed (or faxed in some cases) to each individual dealer. After receiving their statements, the dealers faxed, e-mailed or called their representative to request order changes or corrections.

The dealer representatives, in turn, used the AR accounts receivable application to post and keep track of the change requests and payments. This application connects to the SAP system and reads accounts receivable information from the appropriate SAP database tables, making that information available to the dealers through the company application portal. It also sent change requests to be fulfilled by the SAP system. The AR application therefore acted as an auxiliary tool to aggregate all relevant information to the dealer representatives, avoiding the necessity for the dealer representatives to have extensive SAP training and/or time consuming requests to SAP users.

This process was labor intensive both on the dealers and their representatives. It was also inefficient, since it depended on dealer representatives taking action related to the dealer requests. In times when many such requests occur simultaneously, the dealer representatives constituted a process bottleneck.

When dealer representatives became overloaded with order change requests, the response time to fulfill the requests decreased. This was of particular concern to the company, since it ended up delaying payments. A customer survey also pointed out that dealers thought the process to change orders was complicated. As the majority of dealers work with other manufactures, they can naturally take their business elsewhere. As a result, the company estimated that a significant amount of business was being lost due to those issues related to the accounts receivable process.

9.4 The ECM solution

The ECM solution consisted of two basic components (Figure 9-1):

- Content Manager OnDemand (CMOD), used for report archiving and distribution
- CommonStore for SAP (CSSAP) used for SAP data archiving.

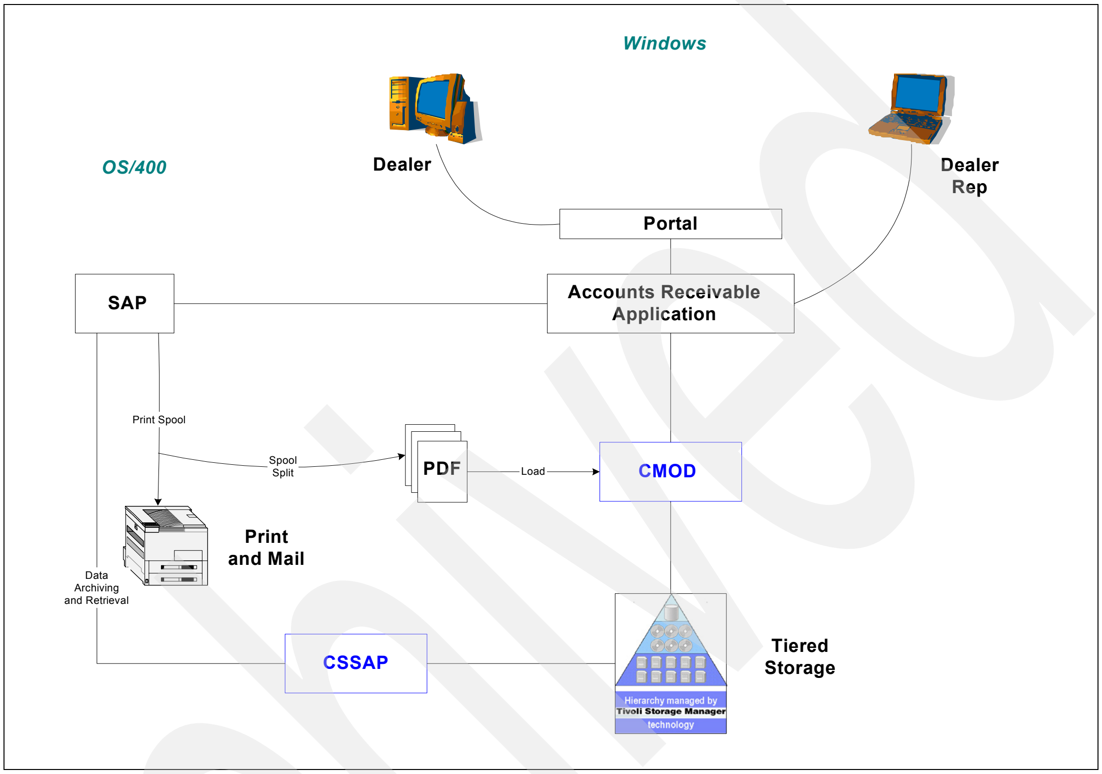


Figure 9-1 ECM Solution Diagram Components

Figure 9-1 displays the solution diagram components and access points. The SAP Enterprise Resource Planning (ERP) system runs on a OS/400® platform, and generates the activity statements for each dealer at the end of every month. The Accounts Receivable (AR) Application accesses SAP tables that contain account activity information, and makes that information available to dealer representatives.

A significant improvement to the process was to provide access to the AR application to the dealers, via the company application portal. This allowed dealers with self-service capabilities to place their own change requests. This in turn alleviated the work load on the dealer representatives, allowing them to concentrate on more strategic issues related to the orders and the relationship with the dealers.

Note that the process of allowing dealers access to the AR application could have been done without using ECM components. However, in order to make the process more complete and satisfying to the dealers, the company decided that it was also important to make the statements available to the dealers online. This was possible using the CMOD component, as explained next.

As mentioned before, the dealers were able to log into the AR application and place order change requests. These requests were routed to the SAP system, in accordance to the company's business rules. The AR application reads the SAP tables at regular intervals, and when changes are detected, they are copied into the AR application.

After the change request was approved on SAP, a new statement was generated. A print spool of this statement was then “printed” to a PDF file, which was loaded into the CMOD system. Access to the CMOD system was provided via an API to the AR application. As a result, when dealers access their account in the AR application, they also have access to their statements online.

When the statements are loaded in CMOD, the users are notified via e-mail. The loading process also creates appropriate indexes that enable the AR application to present statements in context with the information displayed to the end user. When users choose to open one of the PDF files available to them, the file is open by the Acrobat® Reader in their desktop computer.

The other ECM component, CSSAP, provides appropriate archiving and retrieval capabilities for SAP data. The SAP data is archived according to a policy defined by the company.

Notice that both the customer statements and the SAP data can be ultimately stored on TSM-managed storage. This allowed the company to leverage existing investment in storage, and great flexibility on how to utilize the different types of storage (disk and tape).

More details on the function provided by CMOD and CSSAP are provided in the next two sections.

9.5 Report archiving and distribution

This section describes how CMOD was used to support the report archiving process. The volume of information as well as the storage implications are also discussed.

9.5.1 The CMOD solution for report archiving

CMOD is a content management system especially designed to support computer-generated output, and is able to index and store reports such as the customer monthly statements. It also allows the statements to be made available via a client interface. The accounts receivable application accessed CMOD via provide API interface, thus allowing access to end users that logged into the application.

CMOD possesses some capabilities that make it particularly suitable to computer-generated content (for example, consumer statements generated via print spools), including:

- The ability to automatically index computer-generated content, while breaking it into logical units for efficient storage and retrieval
- A graphical user interface (GUI) that allows administrators to create and configure CMOD applications to index any type of computer-generated content

The reports generation process did not have to be changed, and continued to be done by SAP, for all 2,000 dealers, on a monthly basis. In addition to being printed, the statements were also ‘printed’ to a set of PDF files, which naturally contained all dealers invoices for that month.

An application was created using the CMOD GUI, to properly index the PDF files by appropriate searchable indexes: dealer number, invoice date, and regional sales manager code. The CMOD application was used to guide the ingestion, or loading, of invoices into CMOD.

The invoices were then stored in CMOD, and made available to the dealers via a portlet accessing the accounts receivable application. Upon login to the Portal, each dealer had the ability to view their own invoices, and were naturally not able to see invoices from any other dealer. This restriction was accomplished using the out-of-the-box capability of query restriction in CMOD. The query to the system was controlled by dealer number and regional sales number, which were required of the dealers upon login.

Overall system response time was also of great concern. CMOD provides two features that basically ensure that performance will remain the same even with data and content volume growth:

- **Database segmentation:** The indexes are stored in a database. As the CMOD system grows, it automatically segments database tables. This means for example that for every 10 million (a configurable parameter) rows in the table, a new table is automatically created. As a result, have hundreds of millions of rows and still maintain high performance.
- **Efficient content storage:** The original document loaded into CMOD is broken according to the indexes and stored as separate logical files. These separate files, which are essentially blocks of information, are retrieved when users ask for a particular report. Thus, the retrieval process is not dependent on the overall amount of content stored in the system, and performance remains the same even with system growth.

In addition to API access, CMOD also includes a client for desktop access, and an optional component for Web access, the OnDemand Web Enablement Kit (ODWEK). ODWEK provides access to reports stored in CMOD to Web clients or applications. Applications can access CMOD via ODWEK APIs or other components such as Information Integrator for Content (II4C), which is provided with the system, or Information Integrator Content Edition (IICE), which can be used to support more sophisticated implementations involving data federation.

9.5.2 Report archiving storage estimates

The following directives were used to estimate the storage required for the content and data:

- Yearly amount of data to archive: 7.2 GB. This figure was based on sample PDF reports which averaged 20 KB per page, including the company logo and heading pictures, and on a report size of 10 pages on average per month, for each dealer.
- CMOD database growth of about 15% of the volume of content loaded. This figure was based on the type of indexes to be stored for each report, and the overhead associated with the database data.

Based on the foregoing assumptions, the storage requirements for the reports were estimated and are shown in Figure 9-2.

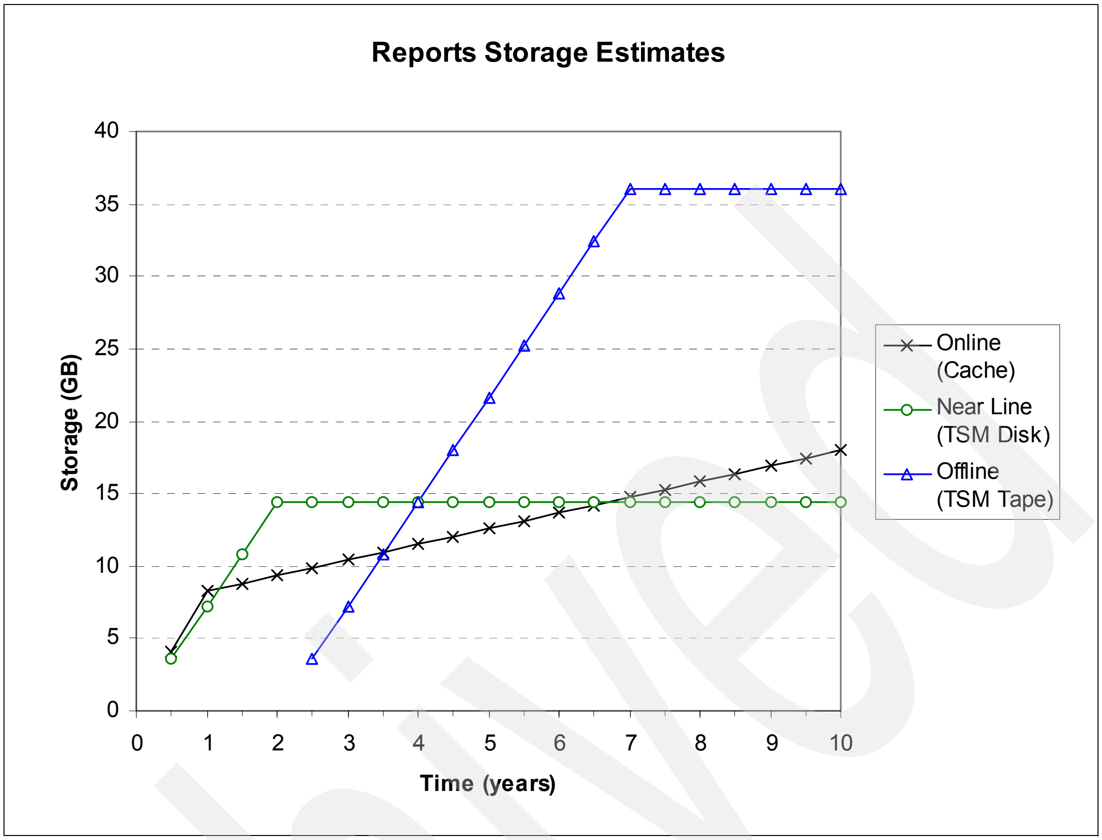


Figure 9-2 CMOD Reports Storage Estimates.

As mentioned earlier, the content is loaded to both the online (cache) and near line (TSM disk) tiers. The cache is configured to expire the documents after one year. The online (cache) storage, as a result, grows fast in the first year as a result of both the cache content and database volume. After that time, cache content expiration starts, and the cache grows solely due to the database.

The second tier, near line storage, grows until the end of the second year, when content migration to the offline storage begins. After that, the near line storage tier reaches a steady-state.

The offline storage, and final tier, starts receiving migrated data from the second tier at the end of the second year and grows until the data reaches 7 years of age. At that point deletion makes that tier reach steady-state.

The online storage is comprised of fast disk attached to the ECM system. However, the ECM system could also leverage network-attached storage (NAS) or storage area network (SAN). These resources were not available to the customer at the time of implementation.

Although the data volumes involved are modest, this case exemplifies how the different storage tiers can and must be leveraged to support an ECM application, while at the same time meeting business and compliance requirements. The example in the next section deals with a significantly larger volume of data.

9.6 SAP data archiving

This section describes the SAP data archiving process and the storage requirements. Particular emphasis is given to the archiving policy definition and its implications.

9.6.1 Archiving policy definition and implications

The archiving policy basically defines what SAP objects must be archived, and for how long. This is an extremely important aspect of archiving SAP data, and it cannot be overly emphasized that definition of the archiving policy is more of a business decision, rather than a purely infrastructure decision. The archiving implications below clarify the requirement for involvement of the company's lines of business on the formulation of the archiving policy:

- **Regulatory Compliance:** This includes compliance with internal policies and external mandates. Internal policies are typically determined by the company's tax department. Internal Revenue Service (IRS) has specific requirements on what business data must be made available during audits. The SAP archiving process creates archives that can be later retrieved and viewed, but with some restrictions. One of the restrictions is the inability of SAP users to drill down on the data. This must be taken into account during the archiving policy definition, and if the company decides that the drill down feature is necessary on the archived data, third-party tools that replace the SAP-provided archiving process must be considered.
- **Backup and Recovery:** SAP database data is part of the production system and therefore backed up on a regular basis. However, once some of the data is archived, it is no longer part of the regular backup process, and therefore steps must be taken to ensure that a proper backup of the archived data is created.

To address the regulatory compliance issue, the company's tax department was involved in the formulation of the archiving policy. A demonstration of the retrieval process was performed to those business users, allowing them to see how the SAP archived objects looked on the SAP GUI after retrieval. Also shown were screen shots of a third party solution that enables drill down on the archived data. After considering previous IRS audit requirements, the tax department personnel decided that the SAP-provided archive was acceptable. However, this might not be the case in other companies and therefore that line of business must always be part of the formulation of the archive policy.

The backup and recovery issue was addressed by leveraging the company's earlier investment in a Tivoli Storage Management (TSM) infrastructure. That infra-structure was already being utilized for backup purposes, mainly for the production databases, such as SAP, as well as for several file and print systems. It was decided that, once the SAP data is migrated to TSM-managed storage, a copy group of the data was made and stored on tape. As part of the backup and recovery policy at the company, an additional copy of tape records was taken regularly and stored in a physical location different from the main hosting site. As a result, this set of measures provided not only adequate backup, but also disaster recovery to the archived data.

9.6.2 Archiving policy

Given the foregoing considerations, the archiving policy was defined for SAP financial data in the following manner:

- During the first six months, data resides in the SAP database.
- After six months, data is archived and moved to disk storage, to remain there for 18 months.

- After 2 years, the data is moved to tape, to remain there for 5 years.
- After 7 years, the data is deleted.

The decision to retain archived data on disk for 2 full years was aimed at providing better response time. The data remains on the SAP database during the first 6 months, after which period it is archived and moved to TSM-managed disk. This addresses two business issues:

- By moving the data out of the SAP system after 6 months, the SAP database growth is controlled, helping avoid system response time degradation
- The data is being moved from a more expensive (OS/400) production system, to a less expensive platform (Windows).

The use of tape as the next storage tier, after the data is 2 years of age, was decided after an evaluation of the response time for data retrieval from the SAP client. The response time, between 2 to 5 minutes, was deemed acceptable by business users since data of that age is used very infrequently. Also considered was the possibility that some SAP archived objects might require data that resides on different tapes. In this case, all tapes will have to be properly loaded before the data is available.

Data protection was ensured with a second copy of archived data made to tape and sent to offsite vaulting on a daily basis. As such, tape copies of two sets of data were made: the data on Windows system disk, which is between 6 months and 2 years of age; and the data on tape, which is between 2 and 7 years of age.

After the data reaches 7 years of age, the deletion process takes place. The process implemented keeps records of the location and content of all tape drives, including the ones shipped for outside storage. It is part of the process to track and ensure that, once a tape is slotted for deletion, both the on-site and off-site copies are deleted.

9.6.3 Storage estimates

Based on the archiving policy described in the previous section, actual storage requirements could be determined. The following assumptions were used to estimate the amount of storage related to each of the storage tiers available:

- Initial size of database: 2 TB, of which 1 TB is eligible for archiving
- Yearly growth of SAP database: 0.4 TB
- Residence time of data on SAP database before archiving: 6 months
- Residence time on near line storage tier (TSM-managed disk): 18 months
- Residence time on offline storage tier (TSM-managed tape) before deletion: 5 years
- Percentage of SAP data archived:70%. This figure depends on the type of SAP data being archived and reflects the fact that some reference information related to the archived transactions must remain in the database.

In addition, a compression rate of 2:1 was assumed for the type of data being archived. Depending on the SAP data being archived, this rate can be higher. The storage requirements for the 3 separate tiers were estimated for the next 10 years of business, and are shown in Figure 9-3.

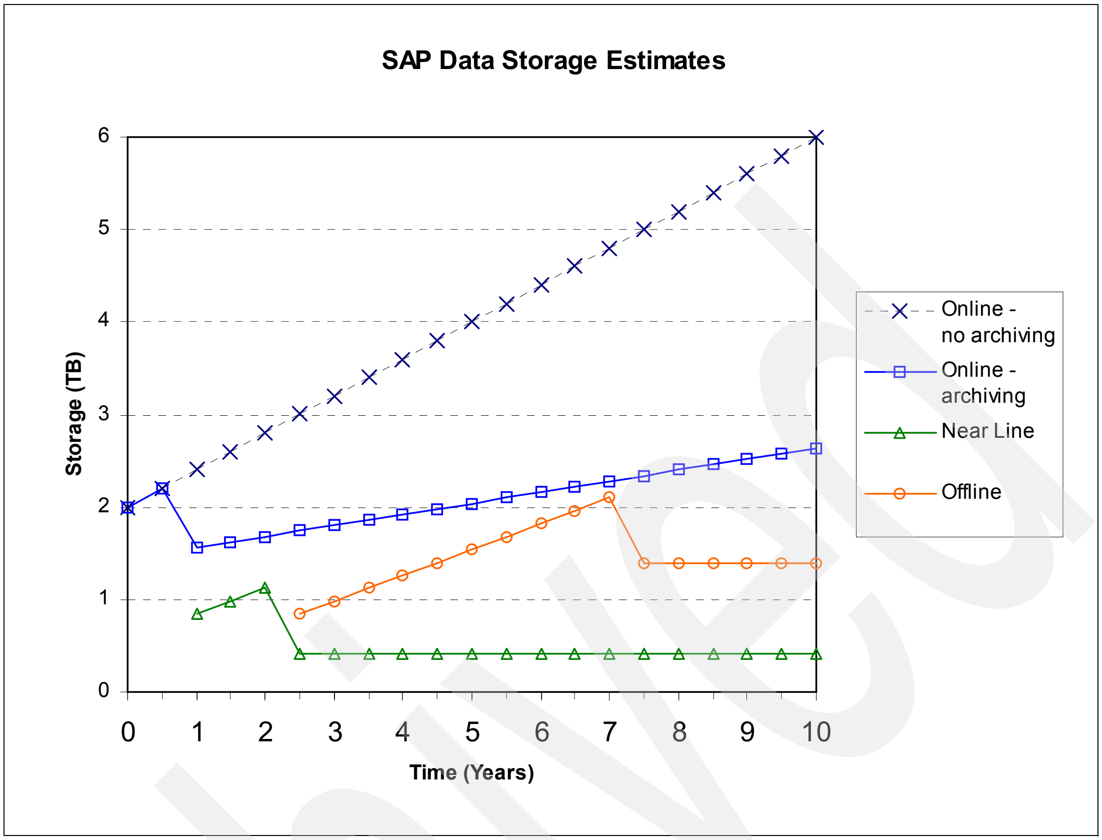


Figure 9-3 Estimates for SAP data archiving

The SAP database, or online tier, grows at a rate of 0.4 TB per year. If no archiving is implemented, the SAP database will reach 6TB on year 10, as indicated by the "Online - no archiving" curve in Figure 9-3.

The archiving process begins by transferring data that is over 6 months of age to the second tier, namely online archiving. The resulting effect on the size of the SAP database is shown in the Figure 9-3 chart by the "Online - archiving" line. That line shows a slower rate of growth as compared to the situation in the "Online - no archiving" line. At the end of 10 years, the projection on the size of the SAP database is about 2.64 TB with archiving, as opposed to 6 TB with no archiving. This size of the database can therefore be kept to less than half of the projected size, with archiving.

The data will reside in the near-line storage (second tier) for 18 months. The storage projection for that tier shows a peak of utilization of1.12 TB on year 2. This estimate takes into account two factors: the compression ration (2 to 1) and an extra copy of the archived data for back up purposes. For simplicity, it is assumed here that the backup copy will remain on the second tier, which is essentially slow-spinning disk. Further storage savings can be obtained by using tape for that copy.

After residing in the second tier, or online archiving, for 18 months, the data is then migrated to the third tier (offline storage). This accounts for the steady-state behavior on the growth of the second tier, which remains at 0.42 TB. In other words, there is no accumulation of data in that tier, since equal amounts are inserted into the tier and subsequently migrated to the next tier.

The offline tier starts out at 0.84 TB between years 2 and 3, which is when the first migration from the previous tier takes place. Due to accumulation, the requirements grow continuously until reaching 2.1 TB at the end of year 7. At that point, data deletion starts, allowing that tier to reach steady-state at 1.4 TB.

Some important aspects are evident in the storage usage in Figure 9-3 on page 180:

- Archiving allows a reduction in use of online storage, which is inherently more expensive (to own, rent, and also maintain), by gradually shifting the data to less expensive tiers.
- The use of different tiers (online, near line and offline) allows great flexibility in meeting several business requirements, while at the same time reducing the cost of operation by leveraging less expensive storage.

Note that the estimates were based on a certain level of business that resulted in a database growth of 0.4 GB per year. This rate, however, tends to grow along with the business volume, so the storage requirements must be re-evaluated accordingly. The increase in business experienced by many companies can be directly correlated with increase demand for storage. By mapping those storage requirements to specific lines of business and storage tiers, companies can better plan their storage usage and future purchases.

9.7 Mapping physical storage to requirements

The overall storage requirements for the data related to the ECM solutions are shown in Table 9-1.

Table 9-1 Minimum storage requirements for the report distribution and SAP data archiving solutions.

Component	Storage Requirement	Storage Characteristics
CMOD System and Logs	12 GB	Internal disk
CMOD Database	10.8 GB	Internal disk
CMOD Cache	18.0 GB	Internal disk
CMOD Data - Near Line	14.4 GB	Fast SAN or NAS attached disk
CMOD Data - Offline	36.0 GB	Fast SAN or NAS attached tape
CSSAP System	2.5 TB	Internal disk
SAP Data - Near Line	1.5 TB	Fast SAN or NAS attached disk
SAP Data - Off Line	2.5 TB	Fast SAN or NAS attached tape

Some components were installed on internal disk given their availability and the small amount of storage involved. However, larger implementations increasingly rely on storage area network (SAN) or network attached storage (NAS) devices.

Several products from the IBM storage solutions can be used to meet all the requirements above. For example, the near line data can be stored on a data storage DS4800 device. The off line data could be stored on a tape storage TS3310 device.

9.8 Disaster recovery and high availability

There are many ways to provide disaster recovery (DR) and high availability (HA) features to applications in general, and ECM systems in particular. The case below is an example of a simple scenario that does not rely on third party solutions such as cluster technology; and yet, it provides sufficient protection and availability for a reasonable cost (see Figure 9-4).

The majority of the company's systems were hosted on Windows platform. SAP and some financial applications are the exception to this rule, being hosted on OS/400. The company has a long-term plan to move its Windows applications to AIX, and once there, leverage High Availability Cluster Multiple Processing (HACMP) high availability features. In the short term, however, the company required some degree of data protection and availability for the ECM system. After an evaluation of costs and benefits available in several scenarios, the company opted for an architecture that is fairly simple, but that meets its DR and HA business objectives.

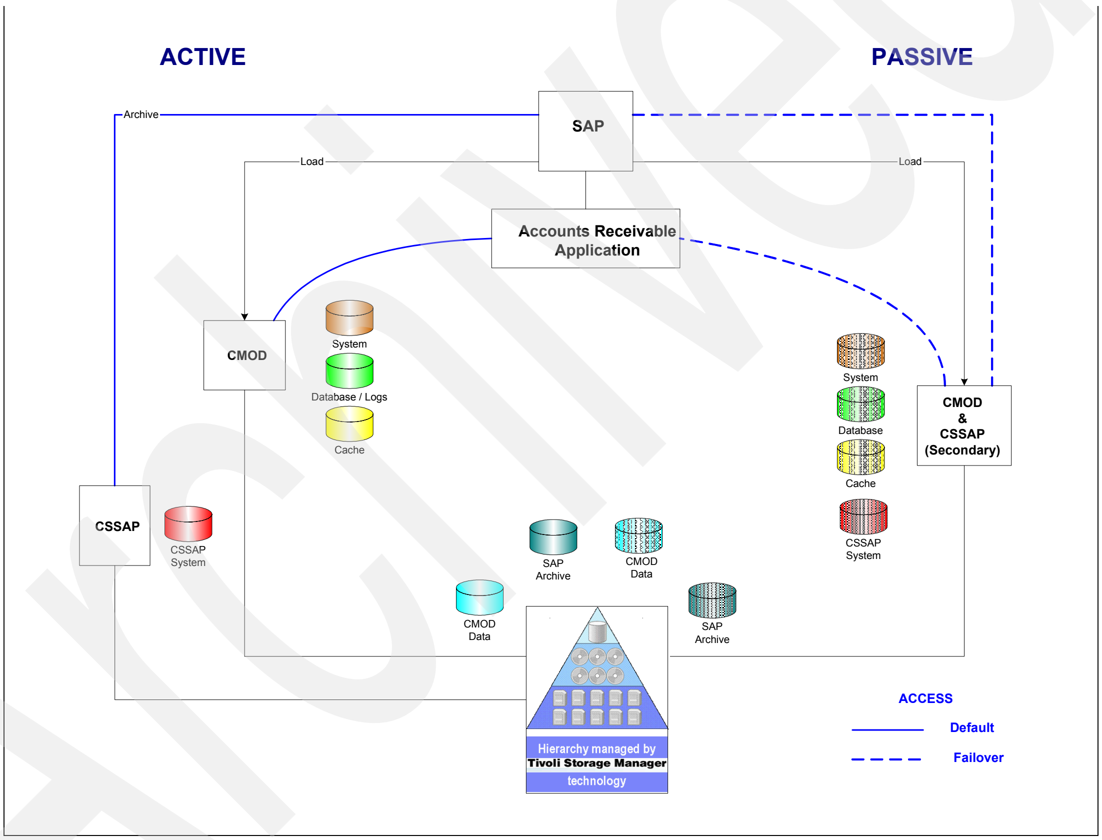


Figure 9-4 ECM high availability configuration.

Figure 9-4 displays the components involved in the DR/HA set up. The DR/HA process is explained for the CMOD and CSSAP components.

9.9 CMOD availability

A secondary CMOD system was created that contained the same configuration as the primary one. The system configuration and the database were the same in both systems. In order to ensure the same objects were loaded to both cache and to a TSM-managed node in both systems, a dual load process was utilized. That way, the same load process was applied to both the primary and secondary systems. Since the load can be treated as a logical unit of work, it is not a complex matter to create an administrative process to ensure that the same load is applied to both systems.

Aliases names were created for both CMOD systems, at the Domain Name Server (DNS) level, for example, *cmod.primary.itso.com* and *cmod.secondary.itso.com*. The DNS entries associate these names with the IP addresses of the actual CMOD servers. The accounts receivable application is configured to connect to the *cmod.primary.itso.com.* In case of failure in that server, the administrator will receive a notification and will switch the alias *cmod.primary.itso.com* to point to the secondary server. Despite being a manual switch over process, the potential down time of up to 30 minutes was perfectly acceptable to the company.

During the load process, the CMOD objects are loaded simultaneously into cache and TSM-managed media. Given that dual load takes place, the cache and 'CMOD objects' components shown in Figure 9-4 on page 182 are therefore identical for both primary and secondary systems. The system configuration components are identical in both systems by creation.

As for the database components, at the end of the business day, the CMOD database is backed up as part of the disaster recovery plan. On Saturdays, the database backup from the primary system is restored into the secondary system. This ensures that potential changes made to the primary database system, such as performance tuning, are reflected in the secondary system as well.

From the foregoing discussion, it can be seen that this setup provides failover availability to the company. It is important to notice that this was achieved with a reasonable cost, without the utilization of cluster technology.

9.10 CSSAP availability

Similarly to CMOD, CommonStore for SAP was also installed and configured in another server. Because CSSAP has such a small footprint, it was decided to install the secondary component in the same server as the secondary CMOD, as shown in Figure 9-4 on page 182.

The primary and secondary systems were given network aliases names, similarly to the CMOD system discussed in the previous section. In case of failure of the primary system, the DNS pointed to the secondary system.

The CSSAP system basically consists of the installed software (binary files), configuration and logs. The logs keep track of what data was archived or retrieved, and in that sense are used for audit or administrative purposes only. For that reason, they are not vital to the functioning of CSSAP, so in case of failure in the primary system, the secondary system can continue to serve the archiving or retrieve requests without the use of the logs from the primary system.

In case of failure of the primary system, it is expected that the secondary system will take over for a period such one business day of work. After that period, and when the primary system is back on-line, the DNS entry will be switched back and point to that system. The log entries from the secondary system will be copied to the primary system, so that the record is kept complete.

9.11 Solution benefits

The ECM solution enabled dealers to access their invoices on-line, and thus verify balance, posting of payments, credit, and any adjustments without having to wait until the following month's statement via mail or fax. As a result, the accounts receivable process was enhanced, and dealer satisfaction was greatly increased.

Additional benefits included more efficient use of storage. Before the ECM implementation, the print spools were kept on the SAP system for several months. After the implementation, the statements were properly filed into their proper storage tier by CMOD, leveraging the TSM storage management capabilities. The statements were loaded on CMOD simultaneously to two storage areas:

- Cache, consisting of fast disk, internal to the CMOD system, and
- TSM-managed storage (node), consisting of (slow) disk.

Caching of data enabled fast access to the most frequently accessed reports. For this company, the most frequent access was of reports that were newer than 60 days. After that period, there was some access for another 30 days, and then access became very infrequent. For that reason, the reports on cache were set to expire in 90 days.

In addition to cache, all reports were also loaded to a storage pool managed by TSM. This pool was set to keep the reports in slow disk for 12 months, then migrate then to tape where they resided for 6 years. Seven years after being loaded into the ECM system, the reports are marked for deletion.

Additional storage-related benefits include native compression capabilities, which for CMOD typically ranges from 2:1 to 30:1, depending on the type of input data. However, since PDF files are already compressed, the compression option was turned off to save resources. For the SAP data archive, however, the estimated compression was around 4 to 1.

Another advantage provided by this solution is in helping achieve regulatory compliance. CMOD captures the data just like it was sent to the printer. In other words, there is no data transformation. As a result, the stored report is a faithful replica of the report that was printed and sent to the end customer. For that reason, CMOD provides a suitable archiving solution that is compliant with regulations requiring the archival of customer statements for a certain length of time. The end result is that the company no longer has to keep paper records of the statements sent to customers. Anytime a particular record is required, it is found in a matter or seconds in a search on the AR application, and, upon retrieval from CMOD, it can be e-mailed, printed or faxed.

10

Case Study III: E-mail archiving

In this case study, we determine the common storage requirements for an e-mail archiving solution. We consider how this solution could be mapped onto the IBM ECM software stack and the IBM Information Infrastructure.

As an example, we design an e-mail archiving solution for ITSO Industries, a company with around 10,000 active e-mail users with 250,000 total e-mail messages per day.

10.1 Introduction

With the increasing use of e-mail as a communication medium for organizations, many e-mails now contain information that must be considered a corporate record, and this now means that e-mail must be proactively managed in accordance with local regulations.

In many cases, the retention period of these corporate records are mandated by government regulation and can vary depending on the type of record.

While these specific regulations differ according to the region or jurisdiction, the *concepts* of compliance and records management remain the same. A corporate record could be a document or e-mail containing results achieved, discussing, and providing evidence of activities performed.

Apart from the pure compliance issues that must be addressed, an e-mail archiving solution also allows an organization's IT department to better manage the e-mail messaging infrastructure by:

- Moving seldom used e-mail messages from the messaging servers to the e-mail archive:

  This allows better utilization of existing systems, which can delay hardware upgrades to the messaging storage infrastructure. Performance is also improved.

- Using single instance storage to reduce storage requirements:

  In most e-mail systems, a full copy of each message is stored in each recipient's "INBOX". For individually addressed messages, this is not an issue, but when large attachments are sent to many users simultaneously, the impact on the messaging system can be overwhelming.

  For example, in an organization of 10,000 people, if each employee were to be sent the same 5 MB attachment, the e-mail servers would require 5,000,000 x 10,000 = 50,000,000,000 Bytes = 50 GB of storage to store that one single message.

  As each of these 10,000 attachments are exactly the same, the use of Single Instance Storage will reduce the storage requirements from 50 GB to 5 MB — a substantial storage saving.

10.2 Business and technical requirements

In this case study, ITSO Industries wishes to implement an e-mail archiving project using IBM software and storage solutions. ITSO Industries is an existing Lotus Domino customer.

In the project, we design and deliver an e-mail archive solution that will allow ITSO Industries to archive all e-mails sent and received by the organization. The solution will implement compliance concepts such as retention policies and legal holds so that e-mails cannot be modified nor deleted if e-mails are subject to legal proceedings.

The solution must allow e-mail users to manage their own mailbox by selectively interacting with the archive solution. The solution must also allow e-mail administrators to implement system wide policies with respect to attachment archiving and e-mail stubbing and deletion from the mail database.

Therefore, the core business requirements for ITSO Industries' e-mail archiving project have been stated as follows:

- Enforce retention policy

- Allow legal holds to be placed on users in the event of litigation
- Keep track of every message going in and out of the company, as well as internal messages among employees
- Enable the legal department to manage e-mail searches and legal holds themselves without depending on IT resources

Disaster recovery must also be addressed in the solution.

10.2.1 Volumetrics

IBM consultants, working with ITSO Industries IT staff, have determined the following volume information regarding the proposed solution and the design and implementation will be based off the following figures.

- Active e-mail users: 10,000
- Number of e-mails per business day: 250,000
- Business days per year: 260
- Average size of e-mail without attachments: 20KBytes
- Average size of attachments: 150KBytes
- Percentage of e-mails with attachments: 25%
- Archiving and indexing of e-mails must be complete before the start of the next business day
- e-mails will be stored on local disk for up to 10 days before being moved to long-term storage
- e-mails must be stored in the archive for 10 years
- Each user will search for and retrieve, on average, 2 old e-mails per day
- Business hours, during which searches and retrievals will occur, are 8am to 6pm
- It has also been indicated that e-mail volumes are expected to increase by 10% yearly.

10.2.2 Functional requirements

Management at ITSO Industries have identified several business problems around e-mail that must be addressed by the solution:

1. For compliance:
   a. All e-mails that are sent or received by the organization must be securely archived.
   b. An e-mail administrator must be able to search for e-mails using full text searching, that is, to search for e-mails and attachments containing words or phrases.
   c. An e-mail administrator must be able to place legal holds on e-mails that meet certain criteria.
   d. An e-mail administrator must be able to extract e-mails from the archive.
2. For mailbox management:
   a. Remove attachments and entire e-mails from users' mail boxes based on a time based policy. The end user can recall the message or attachment from the archive.
   b. An e-mail user must be able to mark an e-mail for archive, retrieval and deletion

10.2.3 Non-functional requirements

ITSO Industries' Lotus Domino servers are running on AIX. The e-mail archiving solution will also run on AIX.

The solution is to be sized for an initial period of 3 years. The solution must allow for additional CPU and storage capacity to be added to cater for the projected increase in mail volume.

ITSO Industries' two data centers are in the same state, with fibre connections between them that can be used by both network traffic and disk replication traffic.

Full text indexing of e-mails must only be performed during an 8 hour overnight processing window so as not to impact searching and retrievals.

ITSO Industries would like the flexibility of adding additional Resource Managers to the solution in the future.

10.3 Designing the solution

The solution that is proposed is based upon DB2 CommonStore for Lotus Domino as the e-mail management product utilizing DB2 Content Manager Enterprise Edition underpinned by storage adhering to the IBM Information Infrastructure.

As this is a compliance solution, we require a software component to control access to previously archived e-mails so that they cannot be modified or deleted. We have decided to utilize IBM e-mail Search for CommonStore as this provides the compliance functionality that is required without having to take the large step of implementing a full blown records management system like IBM DB2 Records Manager.

We will require considerable storage capacity in order to effectively manage 10 years of e-mail and its anticipated growth requirements. We also have to ensure that the storage architecture chosen can accommodate major unforeseen changes in storage and access patterns in the future. The storage must also offer protection to ensure that, at the storage level, the object cannot be modified or lost.

10.3.1 Software

The software components of the proposed solution are as follows:

- DB2 Content Manager Enterprise Edition, which requires:
  - DB2 Enterprise Server Edition
  - WebSphere Application Server
  - DB2 Net Search Extender
  - Tivoli Storage Manager (TSM) / System Storage Archive Manager (SSAM)
- IBM DB2 CommonStore for Lotus Domino, which requires:
  - DB2 Information Integrator for Content (Content Manager API)
  - DB2 Runtime Client
  - Lotus Domino Server (API only)
- IBM e-mail Search for CommonStore, which requires:
  - DB2 Information Integrator for Content (Content Manager API)
  - DB2 Runtime Client
  - WebSphere Application Server
  - Lotus Domino Server (API only)

10.3.2 Proposed architecture

We will utilize DB2 Content Manager as the repository and CommonStore for Lotus Domino as the e-mail archiving component.

At the most basic level we know that we will require a Content Manager Library Server and at least one Content Manager Resource Manager that CommonStore for Lotus Domino will use as its repository. We must also install and configure CommonStore for Lotus Domino on a server. E-mail Search for CommonStore will be installed, in the first instance on the same machine as CommonStore for Lotus Domino but could be moved to a dedicated server at some later stage.

Detailed sizing is given in 10.2.1, "Volumetrics". In summary, we have determined that a single 4-way, 16GB, IBM System p p690 system would approach around 17% of its rated performance before the DB2 Net Search Extender was used to perform the full text indexing of e-mails. To be able to index the 250,000 e-mails during the 8 hour full text indexing windows, we have to be able to index approximately 10 e-mails per second. This is within the capability of DB2 Net Search Extender on this server hardware.

In the smallest installation, we would look at implementing a two-server environment:

1. CommonStore for Lotus Domino and e-mail Search for CommonStore
2. Combined Content Manager Library Server and Content Manager Resource Manager

Figure 10-1 shows a single CommonStore for Lotus Domino server and a single server Content Manager system with collocated Library Server and Resource Manager.

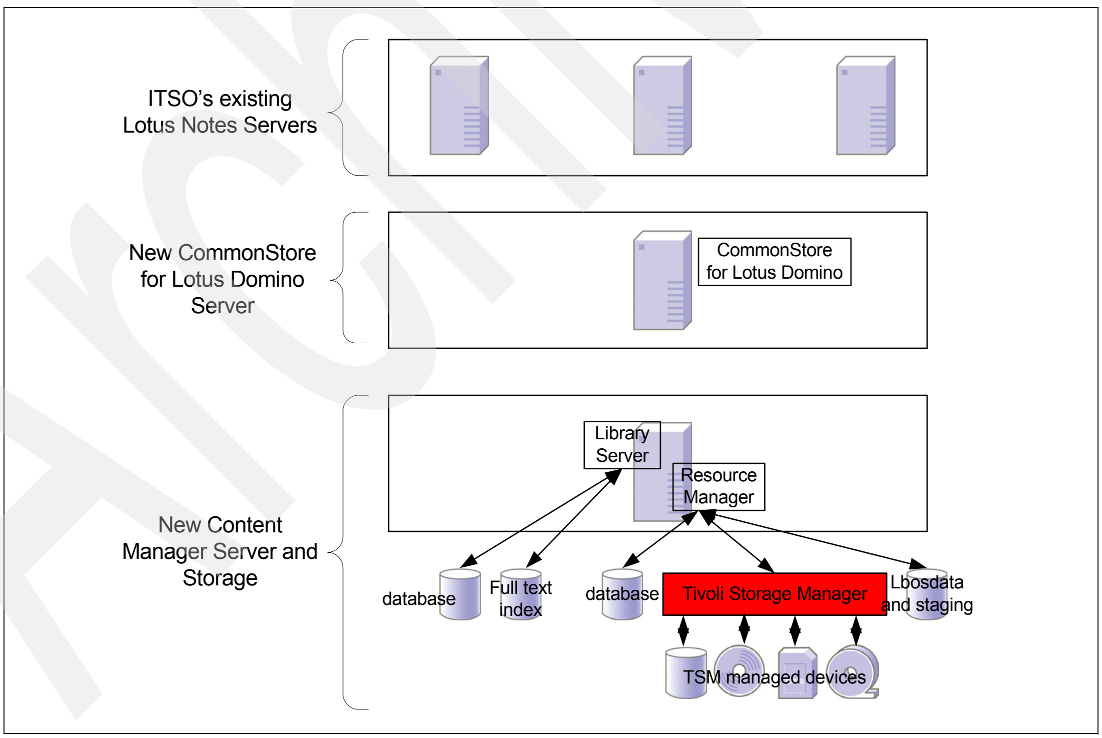


Figure 10-1 Single Content Manager solution with collocated Library Server and Resource Manager

As e-mail volumes and end user usage grow, the underlying hardware would have to be expanded and upgraded and at some stage it will become necessary to separate the Library Server and Resource Manager out onto two machines.

Figure 10-2 shows an expanded solution in which the Content Manager system has been split onto two servers, one server for the Library Server and a second server for the Resource Manager. We have not shown it on this diagram, but additional CommonStore servers could also be added to the system if required.

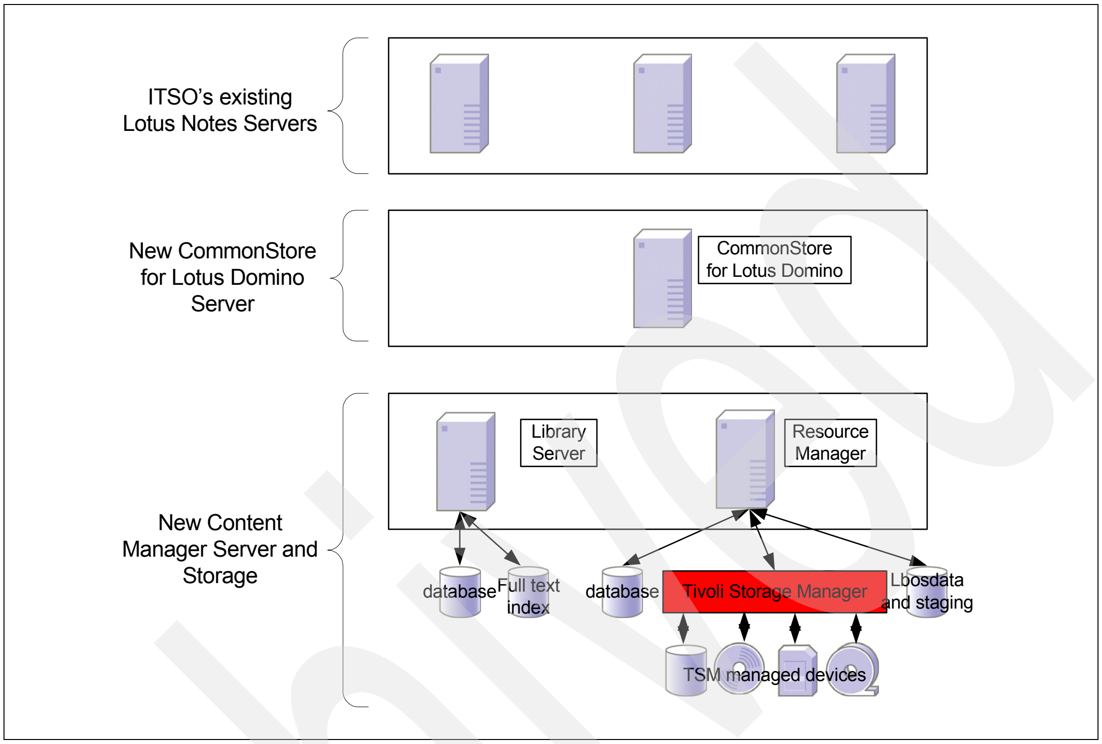


Figure 10-2 Two server Content Manager solution

We also must look at our Disaster Recovery requirements. When a disaster is declared at the production data center, our DR e-mail archiving system, at the DR data center, must be capable of continued operation with minimal disruption and data loss.

In order for this orderly failover to occur, a number of components within the Content Manager and CommonStore for Lotus Domino system must have their data replicated to the DR site.

This data replication between sites could be accomplished by two main methods: first, by using disk replication, or second, by using the replication functionality built into the software applications and middleware. We will explore what data has to be replicated and how this data could be replicated at (a) the disk level, and (b) the application level. Table 10-1 shows replication options for data components.

Table 10-1 Replication options for different data components

Data component	**Replication options**
Library Server database including ► Tablespaces ► Transaction logfiles ► Instance userids	This data can be replicated at either the disk level, or using DB2 High Availability Disaster Recovery (HADR). Other methods such as Log Mirroring and Log Shipping could also be considered for replicating the database.

Data component	Replication options
DB2 Net Search Extender Full Text indexes	These files must be replicated at the disk level. There is no built-in application replication for the Full Text Index files. Other options could include HACMP/XD or IBM Data Mobility Services.
Resource Manager database including ► Tablespaces ► Transaction logfiles ► Instance userids	This data can be replicated at either the disk level or using Resource Manager replication. Other methods such as Log Mirroring and Log Shipping could also be considered for replicating the database. You would choose the same method of replication for the LBOSDATA and possibly Tivoli Storage Manager as you chose for the database.
Resource Manager LBOSDATA	This data can be replicated at either the disk level, or using Resource Manager replication. You would choose the same method of replication for the Resource Manager database and possibly Tivoli Storage Manager as you chose for the LBOSDATA area.
Resource Manager Tivoli Storage Manager data	If you are replicating your resource manager at the disk level, then the Tivoli Storage Manager storage for this resource manager must also be replicated. If you are using Resource Manager replication, then you do NOT have to replicate your Tivoli Storage Manager as the second Resource Manager will usually have its own Tivoli Storage Manager server.

It is also important to remember that at some stage you will have to failback from your DR environment to the Production environment, so the solution chosen must easily support this requirement.

Given the information from Table 10-1, we can see that there are several options for configuring replication in the system.

Library Server database and transaction log replication options are:

- Disk replication
- DB2 HADR replication

For replication of the Library Server database, if you are NOT using Full Text Indexing, either method of replication can be used. However, if you are using Full text Indexing, then disk replication is the simplest solution for replication because it allows all of the DB2 data to be replicated via the same method.

Resource Manager replication options are as follows:

- Database and transaction logs:
  - Disk replication
  - DB2 HADR replication
  - Application Replication

- Objects (LBOSDATA and Tivoli Storage Manager):
  - Disk replication
  - Application Replication

For replicating the Resource Manager, you would usually choose the same method of replication of the database as you do for the e-mail objects.

If you choose to replicate your Resource Manager using disk replication, you will only be able to have one replicated copy of your Resource Manager due to the nature of disk replication. There is a similar restriction if you were to use DB2 HADR to replicate your database. The replicas would also be full replicas of the original data store.

By using the built in DB2 Content Manager Resource Manager replication functionality, you are exploiting the inbuilt flexibility of Content Manager that enables the following capabilities of your objects:

- Selective replication between Resource Managers
- Replication to more than more location (that is, Resource Manager)
- Bi-directional replication
- Scheduled replication

> **Note:** The Resource Manager replication task is *asynchronous*. It could take minutes, if not hours, for objects that have been stored in one Resource Manager to be replicated to other Resource Managers.
>
> This leads to a complication that, if a disaster were to occur, not all objects might be available in the DR Resource Manager even though the Library Server holds metadata about those objects and they would be returned as the result of a search.
>
> One method to mitigate this problem is to configure Content Manager so that objects are initially stored into the DR Resource Manager and then replicated back to the Production Resource Manager.
>
> By doing this we can ensure, that if a disaster were to occur, all objects are available.

Therefore, for this case study, since we require Full Text Indexing to support e-mail Search for CommonStore, we will use disk replication for replicating the Library Server database and full text indexes to the DR site.

For the Resource Manager, since one of our requirements was to be able to add additional Resource Managers into the solution at a later stage, we will choose to use the inbuilt Content Manager replication to replicate objects to the DR site.

Therefore, the architecture of the proposed solution using disk replication for the Library Server and Resource Manager replication for the Resource Manager is as shown in Figure 10-3.

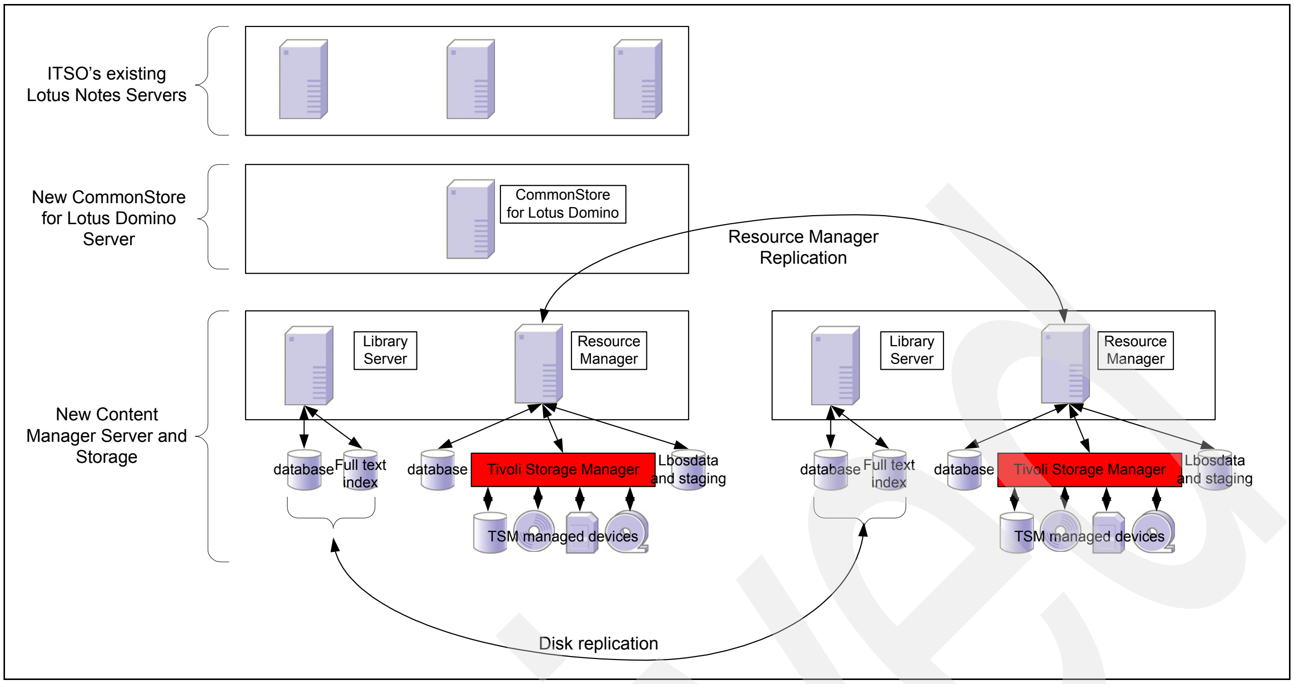


Figure 10-3 Proposed architecture for solution

If we were to add an additional Resource Manager, the solution would look like Figure 10-4. Further, additional Resource Manager's could be added to the system at any time.

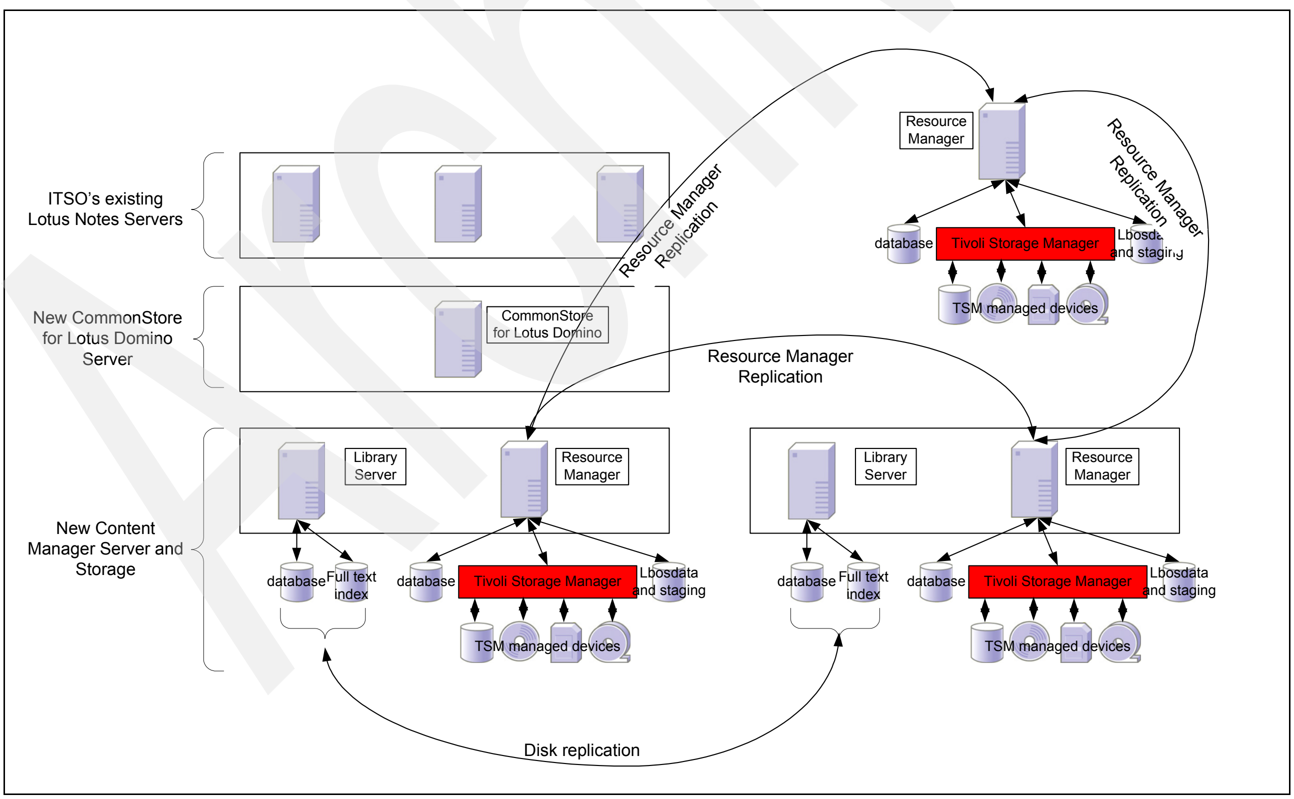


Figure 10-4 Architecture with additional Resource Manager

10.3.3 Storage sizing

From the e-mail volumes in 10.2.1, “Volumetrics”, we are able to determine initial sizing information for the storage required for the system.

There are a number of storage components that must be sized. They include:

- Library Server database and indexes
- Library Server database transaction log files
- Library Server Full Text Indexes
- Resource Manager database and indexes
- Resource Manager database transaction log files
- Resource Manager LBOSDATA storage and staging areas
- Tivoli Storage Manager database
- Tivoli Storage Manager recovery logs
- Tivoli Storage Manager storage pools

Only when we have identified all of the storage that is required to support the solution, can we look at the physical storage options.

Library Server database and indexes

The Content Manager Library Server is the component responsible for storing the information pertaining to every object that has been stored in the Content Manager system and is where all searches for e-mails will be performed. This information is known as metadata. The Library Server is also responsible for storing configuration of Content Manager and also access control information regarding each stored object.

The Library Server is resident within the Library Server database, typically a DB2 database.

In most Content Manager systems, the size of the Library Server database is overwhelmingly determined by the number of e-mails that have been archived. This means that the size of the database will grow in direct proportion to the number e-mails archived for the first 10 years. After ten years, the archive will still continue to grow but at a reduced rate due to old e-mails being purged from the system.

Database indexes are required for efficient searching of the metadata and this information will also grow as the number of objects in the system increases.

Based on the e-mail volumes from 10.2.1, “Volumetrics” on page 187, an estimate for the size of the Library Server database over the first three years is summarized in Table 10-2.

Table 10-2 Library Server database sizing for years 1 to 3

	Year1	Year2	Year3
e-mails stored per year	65,000,000	71,500,000	78,650,000
Total e-mails stored at year end	65,000,000	136,500,000	215,150,000
Database growth per year	194 GB	213 GB	235 GB
Database size at year end	194 GB	407 GB	642 GB

Note: By default, when the Content Manager Library Server database is created, it will be created using SMS (System Managed Space). In some instances, it might be advantageous to use DMS (Database Managed Space) because this allows you more flexibility in the layout of parts of the database.

Your DB2 administrator can provide guidance on whether you ought to be using DMS for your database, as it is very dependent on your requirements and circumstances.

Library Server database transaction log files

All databases require transaction logs. These logs keep a record of the changes that have been applied to a database. These logfiles are used to roll forward a restored database to the point in time of the failure and to wind back transactions that have failed.

In DB2 there are two type of transaction logging, and each provides a different recovery capability.

- Circular logging:

  Circular logfiles are only useful in ensuring the integrity of the current transaction. Circular logfiles are not able to be used to roll forward a restored database to a point in time as old logfiles can be overwritten at anytime. As the old logfiles are automatically reused by DB2, no log file management or archiving is required.

- Archive logging:

  Archive logging, on the other hard, can be used for roll forward recovery. This allows you to roll forward a database after a database restore or a failure to a particular point in time.

As one of our key requirements is to be able to continue working from our DR site after a failure at the production site, we require a logging method that will allow us to recover most, if not all, transactions that have occurred since the last backup up to the point of failure. Therefore, we must ensure that our database environment utilizes archive logging and has sufficient space to store the archive log files and that the logfiles are made available at the DR data center almost immediately.

By default, each DB2 transaction logfile for Content Manager is set to 1000 4KByte pages (that is, 4 MB), and the system is configured with 10 primary and 20 secondary logfiles. These values set the maximum size of the disk storage required for the transaction logs.

The formula to determine the disk space required for the primary and secondary logfiles is:

```
totallogsize = (logprimary + logsecond) * logfilsiz * pagesize
```

Therefore, in a standard Content Manager system we will require the following storage for the transaction logs:

```
totallogsize = (10 + 20) * 1000 * 4096 Bytes
totallogsize = 122,880,000 Bytes = ~ 122 MB
```

Transaction logfiles must be stored on fast storage and have to be replicated to the DR site.

The long-term storage of the archived logfiles must also be considered, as the logfiles could be required when a disaster occurs. Typically, the logfiles are archived to other media or storage locations once DB2 has completed all units of work within the current logfile. It is important to ensure that these logfiles are available to the database server in the DR data center so that they can be used during the recovery process if required.

In our case study, given the e-mail volumes, it would not be unreasonable for several hundred logfiles to be created during a 24 hour period. Therefore we could require in the vicinity of several hundred MB of long-term storage for archived transaction log files per day. This could be disk storage or Tivoli Storage Manager managed storage.

The transaction logfile parameters must be set according to standard DB2 tuning recommendations.

Therefore, for our case study, we require around 130 MB of fast, protected and replicated disk to store the primary and secondary logfiles. We also require archive space, possibly within Tivoli Storage Manager, for the storage of the archived transaction log files.

Full text indexes

The e-mail Search application uses the capabilities of DB2 Net Search Extender to build a full text index of the contents of the archived e-mail messages. These indexes are not stored in the DB2 database, but in a separate storage location on disk, but are still integral to the operation of the system.

The size of the full text indexes can be in the vicinity of 70% of the size of the indexed documents. We also require temporary storage that could be four times the size of the batch to be updated.

During the indexing and merging phases of the Net Search Extender processing, very large amounts of disk based data are manipulated. To allow for processing to occur in a timely fashion the Full Text Indexes must be on extremely fast disk. This not only helps the indexing phase but also search.

It is also advantageous for the temporary files and the actual Full Text Index files to be on the same filesystem. If the files are on the same filesystem the merging phase can simply rename some of the files rather than copy the files (which would be required if the files were on different filesystems).

Given the e-mail volumes from 10.2.1, "Volumetrics" on page 187 based on an average document size of 52,500 Bytes, it would be reasonable to assume the full text index will use 36,750 Bytes per e-mail. This is summarized in Table 10-3.

Table 10-3 Estimated Full Text Index size for first three years

	Year 1	**Year 2**	**Year 3**
e-mails stored per year	65,000,000	71,500,000	78,650,000
Total e-mails stored at year end	65,000,000	136,500,000	215,150,000
Full Text Index growth per year	2,388 GB	2,627 GB	2,890 GB
Full Text Index size at year end	2,388 GB	5,016 GB	7,906 GB

It is worth noting here that the system administrators have to monitor the performance of the Full Text Indexing tasks to ensure that the time taken to perform the overnight indexing and the occasional maintenance tasks does not exceed the available processing window.

In "*Best Practices for Setting Up an IBM CommonStore Solution For Mailbox Management, e-Mail Retention and Discovery*, SG24-7325, it is recommended that the size of Full Text Index does not exceed 1 TB.

One proactive solution is to configure the system so that, rather than have a single large Full Text Index, we create several smaller indexes based upon the date the e-mail was received in the users “INBOX”. For example, we could create an index for each Quarter in the year.

The e-mail Search for CommonStore application supports searching across multiple indexes split via date to support the 1 TB recommendation.

Resource Manager

The Content Manager Resource Manager component manages the storage of the actual e-mail. It is responsible for determining whether the object ought to reside in the LBOSDATA area, or whether it ought to be handed off to Tivoli Storage Manager to manage the object.

A database is required to support the Resource Manager, but it is small in comparison to the size of the Library Server database and of the e-mails themselves.

The major storage requirement is for the storage of the e-mail objects.

Resource Manager database and indexes

The Content Manager Resource Manager is the component responsible for managing and recording the physical storage location of every object that has been stored in the Content Manager system.

The Resource Manager is a Java application running within a WebSphere Application server that uses the Resource Manager database, typically a DB2 database, to keep track of objects.

As with the Library Server, the size of the Resource Manager database is overwhelmingly determined by the number of e-mails that have been archived, meaning that the size of the database will grow in direct proportion to the number e-mails archived for the first 10 years. After ten years, the archive will still continue to grow but at a reduced rate due to old e-mails being purged from the system. Similarly, the size of the database indexes will grow as the number of objects stored in the Resource Manager increases.

Based on the e-mail volumes from 10.2.1, “Volumetrics” on page 187, an estimate for the size of the Resource Manager database over the first three years is summarized in Table 10-4.

Table 10-4 Resource Manager database sizing for years 1 to 3

	Year1	Year2	Year3
e-mails stored	65,000,000	71,500,000	78,650,000
Total e-mails stored at year end	65,000,000	136,500,000	215,150,000
Database size	39 GB	43 GB	48 GB

Remember that we will not be replicating the Resource Manager database at the disk level but rather at the application level, and the Content Manager administrator is responsible for choosing what subset of the objects must be replicated to other Resource Managers.

Resource Manager transaction logs

As with the Library Server database, the Resource Manager database must be configured to use archive logging so that we can recover to a point in time after a database restore or recovery.

The default values for the database transaction logs upon Resource Manager creation are:

- logprimary = 25
- logsecond = 50
- logfilsz = 1000

Ac the formula, the disk space required to hold the primary and secondary logfiles is:

```
totallogsize = (logprimary + logsecond) * logfilsiz * pagesize
```

Therefore, in a standard Content Manager system we will require the following storage for the transaction logs

```
totallogsize = (25 + 50) * 1000 * 4096 Bytes
totallogsize = 307,200,000 Bytes = ~ 307 MB
```

The active log files must be stored on fast, protected storage. This disk does not have to be replicated to the DR site as we are using application replication to replicate the e-mails to the DR data center.

As with the Library Server, once all the units of work in the active logfile have been completed, the log file will have to be archived. This archival storage could be either disk storage or Tivoli Storage Manager.

For our case study, it would not be unreasonable for the Resource Manager to utilize several hundred logfiles during a 24 hour period which have to be stored on protected storage for possible later use.

Resource Manager LBOSDATA and staging

The LBOSDATA area is fixed disk storage local to the Resource Manager machine that is directly under the control of the Resource Manager.

Objects that are stored in LBOSDATA will have faster retrieval time than if the object was stored in Tivoli Storage Manager. This is very useful in situations where significant work is performed on objects that have been newly archived, for example, the full text indexing phase.

Objects are migrated from the LBOSDATA area primarily based on time based rules but can also be migrated based on disk threshold settings. The movement of objects from LBOSDATA to Tivoli Storage Manager is performed by the Resource Manager Migrator asynchronous task. When the Resource Manager is configured, the Migration Policy, instructs the Resource Manager and the Migrator task

- Where objects reside
- How long they reside in that particular storage area
- Where they must be migrated to after the time period expires

As an adjunct to the time based migration, the system administrator also tells the Resource Manager and Migrator that if LBOSDATA reaches a certain threshold, objects must be migrated to their next storage location to free up space in LBOSDATA. This will ensue that newly archived objects will have space available for them to be stored in LBOSDATA.

For our case study we will store e-mails into LBOSDATA for 10 days and then migrate to long-term storage.

The staging area is local disk that is used as a cache area for objects that have been retrieved from Tivoli Storage Manager. Just like the LBOSDATA areas, it is faster to return objects from local storage than from Tivoli Storage Manager.

The parameters of the staging area are controlled by the system administrator who can set:

- Amount of disk space that the staging area can use
- Upper threshold over which cached e-mails that are not frequently used will be purged by the Resource Manager Purger asynchronous task
- Lower threshold where the Resource Manager Purger asynchronous task will stop purging e-mails from the staging area

Objects that have been purged from the staging area are still retrievable, but they will have to be retrieved from TSM on the next access.

It would not be unreasonable to size the staging area to hold one day of retrieved objects but the solution architect has to study the retrieval patterns of the e-mails to confirm the ideal size of the staging area. For our case study, we will size the staging area to hold one day of retrieved e-mails.

Table 10-5 Resource Manager LBOSDATA and staging sizing for years 1 to 3

	Year1	**Year2**	**Year3**
e-mails stored	65,000,000	71,500,000	78,650,000
Total e-mails stored at year end	65,000,000	136,500,000	215,150,000
e-mails retrieved daily	20,000	22,000	24,200
LBOSDATA size	141 GB	155 GB	170 GB
Staging area	1 GB	1.15 GB	1.3 GB

Tivoli Storage Manager

For storage of objects to non LBOSDATA disk storage, the Content Manager Resource Manager uses Tivoli Storage Manager. Tivoli Storage Manager allows Content Manager to utilize any of the large number of storage devices that Tivoli Storage Manager supports.

> **Note:** When we say Tivoli Storage Manager (TSM) we are also including System Storage Archive Manager (SSAM).
>
> SSAM is a version of TSM that offers Data Retention functionality. To use functionality such as nSeries Snaplock, SSAM is required.

For example, we could use the following storage devices

- Tape
- Optical / DVD / CD libraries
- NAS
- SAN
- Content Addressable Storage
- Date retention devices

Tivoli Storage Manager can also make internal backup copies of objects on different storage devices, to protect against data loss in the event of failure of a storage device. For example, you could configure Tivoli Storage Manager to store the primary object copy on N series disk, and to keep a backup (in a copy storage pool), on WORM tape.

As new devices are supported by Tivoli Storage Manager existing objects can be migrated from old devices to new devices transparently - without requiring intervention from Content Manager.

Ultimately, the physical storage location of the object is unknown to the Content Manager system. Content Manager only has to know that the object is managed by Tivoli Storage Manager - Tivoli Storage Manager then manages the actual physical storage location.

For our case study, since we require long-term protected and compliant storage, we will utilize Tivoli Storage Manager to manage the objects that have been migrated from LBOSDATA.

Much like the Resource Manager itself, we have to provision storage for the Tivoli Storage Manager database, transaction logs and the storage pools that will store the actual e-mails.

The relationship between a Resource Manager and a TSM server can be a many-to-many relationship. A Resource Manager is not limited to using a single TSM server, and a single TSM storage can store objects from many Resource Managers. For the purposes of our case study we will assume that a single Resource Manager talks to a single TSM server and a TSM server only services a single Resource Manager, a one-to-one relationship.

Tivoli Storage Manager database

Each object stored in Tivoli Storage Manager will have an entry in the Tivoli Storage Manager database. On average, each object stored into Tivoli Storage Manager will use ~500 bytes in the Tivoli Storage Manager database. The size of the database will increase linearly as the number of objects stored increases.

When Tivoli Storage Manager databases grow very large, performance can start to degrade. It is generally recommended to limit a Tivoli Storage Manager database to less than 120 to 150 GB.

To alleviate this issue and to allow Content Manager to store large volumes of objects to Tivoli Storage Manager, the Content Manager Resource Manager allows the system administrator to enable “aggregation”. When aggregation is used, Resource Manager objects are batched together into one Tivoli Storage Manager object, thereby reducing the overhead on the Tivoli Storage Manager database. Note that you can only use aggregation when the Resource Manager is using Tivoli Storage Manager in “archive copy group mode,” which is also known as standard retention mode.

To enable retention and aggregation mode, the Tivoli Storage Manager device driver within the Content Manager Resource Manager configuration must have its parameters set to:

```
mode=retention_aggregate
```

Also, the VOL_AGGREGATESIZE column value of the LBOSDATA volumes in the RMVOLUMES table in the Resource Manager database must be set to the aggregation size that must be used. That is, if you wish to aggregate objects in LBOSDATA into 10 MByte objects within Tivoli Storage Manager then the VOL_AGGREGATESIZE value must be set to 10,000,000 against the LBOSDATA volumes the objects will be migrated from. The maximum number of Resource Manager objects that will be aggregated into one Tivoli Storage Manager object is 200 regardless of the value of VOL_AGGREGATESIZE.

Therefore, for our case study, as our average object size is 52.5 KBytes, with an aggregation size of 10 MB, the Resource Manager will store up to ~190 objects in the aggregated Tivoli Storage Manager object and consequently our database size will be reduced significantly.

There is, however, a drawback from using an aggregation size that is too large. When the Resource Manager has to retrieve an object the entire aggregated object has to be retrieved by the Resource Manager before the actual e-mail can be extracted from the aggregated object which can then be returned to the client application. This behavior adds a small overhead to each object retrieval, therefore the aggregation size has to tuned for each project. This is summarized in Table 10-6.

Table 10-6 Tivoli Storage Manager database sizing for years 1 to 3

	Year1	**Year2**	**Year3**
e-mails stored	65,000,000	71,500,000	78,650,000
Total e-mails stored at year end	65,000,000	136,500,000	215,150,000
Tivoli Storage Manager database size without retention aggregation	32.5 GB	68.25 GB	107.5 GB
Tivoli Storage Manager database size with retention aggregation	171 MB	359 MB	566 MB

See the Content Manager Information Center on ibm.com for more details in object aggregation.

IBM recommends that the Tivoli Storage Manager database be mirrored using the inbuilt Tivoli Storage Manager database mirroring capabilities. The database mirror must use a different disk than is used for the primary copy of the database.

Given the mirroring requirement, disk storage requirements for the database are double the sizes indicated in Table 10-6.

Tivoli Storage Manager recovery logs

The Tivoli Storage Manager recovery log is Tivoli Storage Manager database's transaction logs.

There are two modes of recovery logging in Tivoli Storage Manager.

1. Normal logging:

   Like DB2's circular logging, Tivoli Storage Manager's normal logging is used to keep track of in-flight transactions and cannot be used to recover to a point in time after a Tivoli Storage Manager database restore.

2. Roll forward logging:

   Roll forward logging is synonymous with DB2's Archive logging in that logfiles are not reused after a commit. They are retained too be used for recovery after a Tivoli Storage Manager crash or database restore.

To estimate the disk space required for the recovery logs it is recommended that the Tivoli Storage Manager log usage be monitored while the system is being used under its normal workload. By monitoring the log consumption, the Tivoli Storage Manager administrator will be able to allocate sufficient initial log storage as well as configuring maintenance tasks to automate tasks such as:

- Recovery log extensions
- Database backups

IBM recommend that the recovery logs be mirrored using the inbuilt Tivoli Storage Manager recovery log mirroring functionality. The mirror copies of the recovery logs must use a different disk than is user for the primary copy of the recovery logs.

For the purposes of this case study[1], we will roughly estimate that the initial size of the Tivoli Storage Manager recovery logs will be ~150 MB (250,000 e-mails x 500 bytes/e-mail + overhead) per day. To allow for growth and unforeseen increases in traffic, we will allocate 500 MB to the primary recovery logs. A similar amount of disk space must be allocated to the mirrored copy of the recovery logs.

Tivoli Storage Manager storage pools

The Tivoli Storage Manager storage pools are where the actual e-mail objects, or aggregated e-mail objects, and other objects are stored.

For our case study, we have already identified a number of different objects that have to be stored to Tivoli Storage Manager for medium and long term management. Some of the objects require storage on compliance hardware (i.e Snaplock, DR550 or WORM tape) while other objects do not have their retention managed. Table 10-7 shows the Tivoli Storage Manager storage requirements.

Table 10-7 Different types of objects requiring storage in Tivoli Storage Manager

Object Type	Medium or long term storage	Retention managed
Library Server archive log files	Medium term, 60 days	No
Resource Manager archive log files	Medium term, 60 days	No
Aggregated e-mails	Long term, 10 years	Yes

For the purposes of this case study, we will assume that the underlying storage hardware is resilient with mirroring of data so that we are not required to store two copies of the objects within Tivoli Storage Manager. The storage required for the e-mails is shown in Table 10-8.

Table 10-8 Estimated Tivoli Storage Manager e-mail storage for years 1 to 3

	Year1	Year2	Year3
e-mails stored	65,000,000	71,500,000	78,650,000
Total e-mails stored at year end	65,000,000	136,500,000	215,150,000
Storage required	3,412 GB	7,166 GBGB	11,295 GB

[1] The Tivoli Storage Manager administrator must monitor recovery log usage of the Tivoli Storage Manager server and tune the recovery logs to reflect the actual usage patterns of the system.

This long term storage is per Resource Manager and assumes full replication of e-mails between Resource Managers.

The medium term storage required for the Library Server and Resource Manager database transaction logs files is around 1GB per day per database and would have a small, perhaps 60 day, lifetime.

Note that other objects could be stored in Tivoli Storage Manager such as database backups, Content Manager server file system backups, and so on. These are not included in this case study and would increase Tivoli Storage Manager storage requirements accordingly.

10.3.4 Mapping storage requirements onto physical storage

We have now identified the major storage requirement for our e-mail archiving solution and we now look at how we can map the differing requirements onto IBM storage (Table 10-9).

Table 10-9 Storage requirements and characteristics

	Storage requirement for year 3	**Storage characteristics**	**Comment**
Library Server database	642 GB	Fast SAN or NAS attached disk. Disk must be mirrored and replicated to DR site	
Library Server active transaction logs	130 MB	Fast SAN or NAS attached disk. Disk must be mirrored and replicated to DR site	
Full Text Indexes Indexes: Temporary:	 2 GB 7,906 GB	Fast SAN or NAS attached disk. Disk must be mirrored and replicated to DR site	
Resource Manager database	48 GB	Fast SAN or NAS attached disk. Disk must be mirrored.	Per Resource Manager
Resource Manager active transaction logs	307 MB	Fast SAN or NAS attached disk. Disk must be mirrored.	Per Resource Manager
LBOSDATA	170 GB	Fast SAN or NAS attached disk. Disk must be mirrored.	Per Resource Manager
Staging	1.3 GB	Fast SAN or NAS attached disk. Disk must be mirrored.	Per Resource Manager
Tivoli Storage Manager database	566 MB or 1.2 GB if mirrored by Tivoli Storage Manager	Fast SAN or NAS attached disk. Disk must be mirrored.	Per Tivoli Storage Manager Server
Tivoli Storage Manager Recovery Logs	500 MB or 1 GB if mirrored by Tivoli Storage Manager	Fast SAN or NAS attached disk. Disk must be mirrored.	Per Tivoli Storage Manager Server
Tivoli Storage Manager medium term storage	120 GB	Slow disk or tape; must be protected in the storage layer.	Per Tivoli Storage Manager Server
Tivoli Storage Manager long term storage	11,295 GB	Storage with data retention capability. nSeries with Snaplock or DR550. Storage must be mirrored in the storage device.	Per Tivoli Storage Manager Server

From Table 10-9 on page 203, we can see that we have a requirement for three different types of storage.

First, we require fast storage for database and initial object storage so that e-mail archive and Full Text Indexing are as fast as possible.

Second, the archived DB2 database logfiles that are being stored to Tivoli Storage Manager only have a small chance of ever being read so can be stored on slower storage.

Finally, the largest storage component is required for the storage of the e-mails that have been stored to Tivoli Storage Manager. The storage of e-mails requires retention management so that e-mails cannot be modified or deleted from storage, therefore storage like DR550 or N series with Snaplock is required. See Figure 10-5.

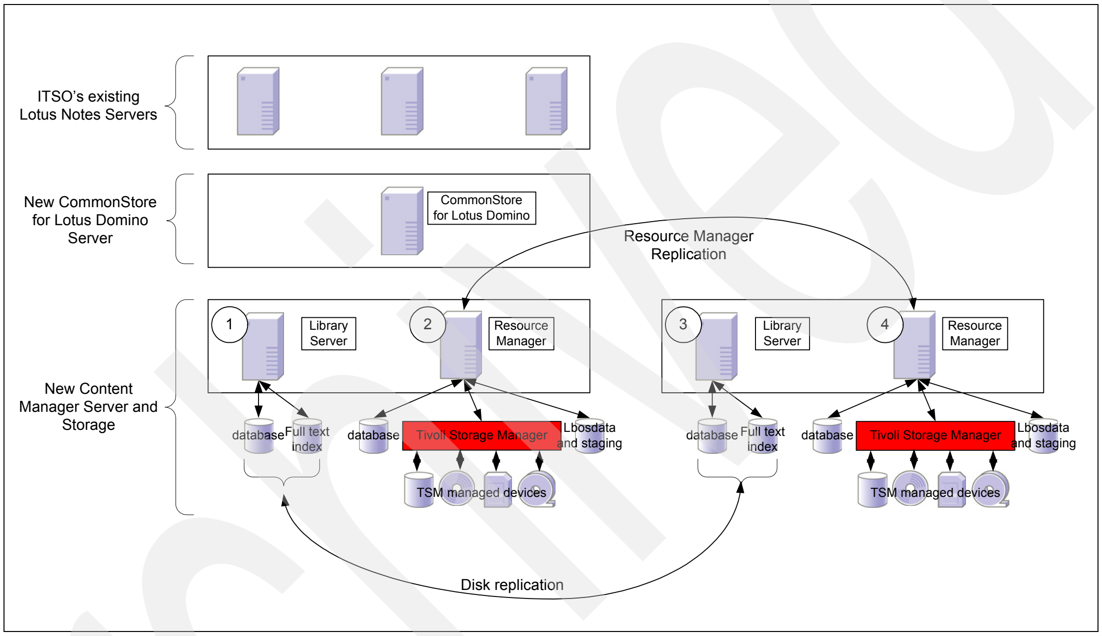


Figure 10-5 Proposed replication solution

If we remember back to our replication requirements, we decided that the Library Server data would be replicated between sites using disk replication. In Figure 10-5, this replication is between servers 1 and 3.

Resource Manager replication, which will replicate objects between Resource Managers, will be performed using the Resource Manager Replicator asynchronous task. This means that the cross site replication in Figure 10-5 between servers 2 & 4 does not require replicated storage.

Figure 10-6 shows one possible solution for our storage using IBM storage solutions. We use storage from the DS family, for example DS8000, for storage of databases, transaction logs, Full Text Indexes, e-mails in LBOSDATA and Staging. The flexibility of the storage allows us to selectively replicate to our DR site.

For long term storage of e-mails we require storage with retention management capabilities. For our solution we could choose either nSeries with snaplock, or we could choose the DR550. Both storage solutions will ensure that e-mails cannot be modified or deleted prior to their designated lifetime expiring.

We have also included additional IBM storage that could be used to store our e-mails on non-disk based technologies for long term storage. The use of WORM tape would allow us to have two unmodifiable copies of an e-mail to mitigate a catastrophic loss of storage.

However, it is important to note that many IBM storage products could have been used to satisfy the requirements for our case study. Chapter 5, “Storage products overview and matrix” on page 69 offers a fuller account of the gamut of storage that we could have used.

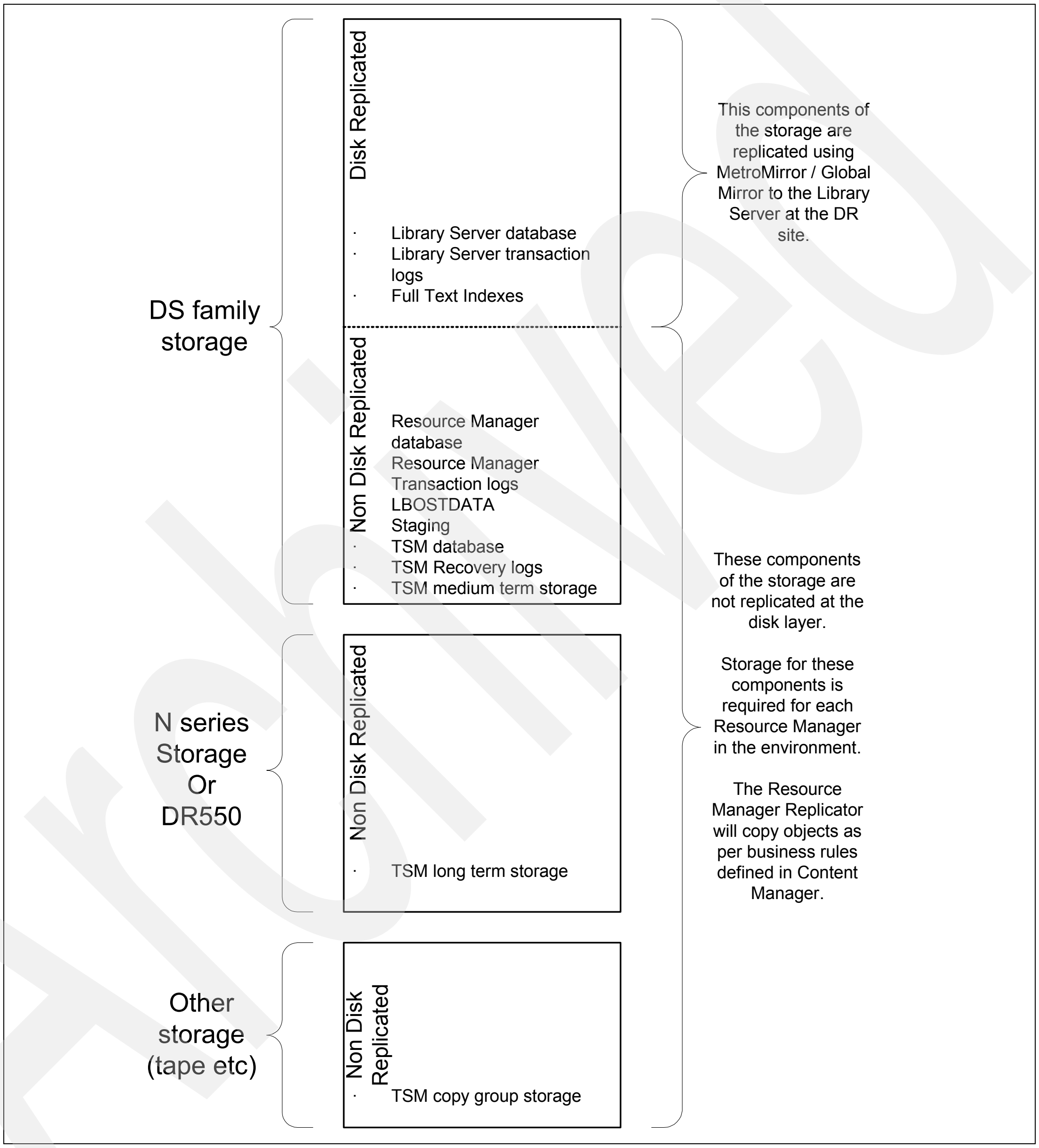


Figure 10-6 One possible storage solution

10.4 Additional comments

It is impossible in a case study to discuss all eventualities. The DB2 Content Manager and CommonStore software is extremely flexible and any given problem can be solved in many ways.

By the same token, the depth and breadth of the available IBM storage hardware would have allowed us to solve our e-mail archiving solution in many different ways.

It is important when building a complex solution that the Solution Architect work hand in hand with the Software and Storage specialists so that all understand the requirements and the decisions that are made as the final solution is designed.

Due to the increasing importance of e-mail and the increasing volumes of e-mail, it is important to design a solution that is expandable, both in terms of processing but in storage.

10.5 Further reading

In this section we give sources for further information on applications discussed in this section.

10.5.1 DB2 HADR

DB2 high availability disaster recovery (HADR) is a data replication feature that provides a high availability solution for both partial and complete site failures. HADR protects against data loss by replicating data changes from a source database, called the primary, to a target database, called the standby.

Further information can be found at:

http://publib.boulder.ibm.com/infocenter/db2luw/v8/index.jsp?topic=/com.ibm.db2.udb.doc/core/c0011585.htm

10.5.2 HACMP/XD

IBM High Availability Cluster Multiprocessing (HACMP) helps achieve business continuity that reliable hardware alone cannot provide. For over a decade, IBM HACMP has provided reliable high availability services, monitoring clients' mission critical applications running on IBM servers—and now the IBM System p and System i™ servers. From the network server operating systems and applications, HACMP monitors the entire system and can quickly and automatically restart an application on backup hardware in the event of failure or service degradation. The combination of IBM's servers, AIX or Linux on POWER™, and HACMP provides the highest level of protection and availability.

And to keep the business running even if an entire site is disabled, HACMP/XD (Extended Distance) extends HACMP's capabilities by replicating critical data and enabling failover to a remote site. HACMP/XD provides a portfolio of data mirroring and recovery options which let you build a disaster recovery solution with the backup site in a campus or metropolitan wide area, or even hundreds of miles away.

Further information can be found at:

http://www-03.ibm.com/systems/p/advantages/ha/

10.5.3 IBM Data Mobility Services

IBM Data Mobility Services helps you address these issues by providing:

- A comprehensive set of services which span the entire lifecycle of data mobility, inclusive of storage infrastructure discovery, assessment, design, plan, implementation and validation
- Services which are inherently non-disruptive to the business, allowing work to be done without impacting the IT production environment
- Services which can be used across storage platforms, operating system environments, and which can be delivered as labor-based or software-based solutions
- Services which are applicable to your most critical projects including: data center relocation, disaster recovery, business continuance, regulatory compliance data protection and archiving, storage consolidation and optimization and Green Data Center initiatives

Further information can be found at:

http://www-935.ibm.com/services/us/index.wss/offerfamily/its/a1029084

10.5.4 Content Manager Information Center

The Content Manager Information Center on ibm.com is the definitive online reference for Content Manager. It can be found at:

http://publib.boulder.ibm.com/infocenter/cmgmt/v8r4m0/index.jsp

11

Case Study IV: Online banking services - Using Image Manager for bank drafts

In a previous case study, ITSO Banking Corporation implemented a turn key online banking solution which provides small banks the capability to offer online banking services at a fraction of their current costs. In a year since their go-live date, they have signed two additional bank partners and the storage foot print has demonstrated the growth expected. Their first year goal was to have eight partners by year end, leaving them two short.

Sales staff at ITSO has communicated that potential partners will close the sale without the ability to display images of checks and money orders online. A survey among current partner banks overwhelmingly revealed that they would be motivated to pay a higher transaction price for the capability to display checks though their online interface.

11.1 Requirements and assumptions

The sales staff at ITSO banking have indicated a requirement to feature imaging capabilities for paper bank notes such as checks and money orders. A survey or existing bank partners supports this position, but the current costs of storing fax images makes ITSO's leadership to be skeptical of its viability. The ITSO business model depends solely on it capability to deliver online banking services at a lower cost than their partners; to remain competitive ITSO must deliver this feature at a storage cost lower than commonly available on the IT marketplace. ITSO has agreed to fund this project only if a viable business model can deliver significantly more storage at a reduced cost per gigabyte.

11.1.1 Determining business requirements

ITSO Banking, since launching their service, has refined their business requirements to better reflect their experiences over the last year and make them more directly applicable to their ECM platform. In addition more specific requirements have been made for supporting the delivery of check images.

- New infrastructure costs be kept in line with revenue. Infrastructure costs which outpace the projected revenue growth are unacceptable.
- Infrastructure changes must not threaten the capability to start service in ten business days of the close of a new contract nor impact bank partners in a negative way.
- Partner banks must be able to offer their customers the ability to display checks, money orders or any other type of paper bank order online.
- While the capability to fax in bank transactions has been successful, the cost of archiving this data relative to it value is prohibitive. ITSO requires a solution which allows this data to be archived for up to seven years at a lower cost.

11.1.2 Determining technical requirements

From the expanded business requirements, further technical requirements have been provided to better define the obligations the imaging solution must meet. Each business requirement has been given an equivalent technical requirement to ensure the direction of the business is reflected in the capabilities required of the infrastructure.

- New infrastructure costs be kept in line with revenue. Infrastructure costs which outpace the projected revenue growth are unacceptable.
  - The potential cost of storing check images over seven years will likely approach hundreds of petabytes. The cost of storage will play a key role in determining the viability of providing this service.
- Infrastructure changes must not threaten the capability to start new service in ten business days of the close of a new contract nor impact bank partners in a negative way.
  - The addition of check imaging must not prolong the implementation date for a new customer. Hardware additions must occur without impacting customers.
- Partner banks must be able to offer their customers the ability to display checks, money orders or any other type of bank order online.
  - Delivery of image data must integrate with the existing infrastructure and programatic delivery of the existing online banking system.

- While the capability to fax in bank transactions has been successful, the cost of archiving this data relative to it value is prohibitive. ITSO requires a solution which allows this data to be archived for up to seven years at a lower cost.
  - The primary expense of storing faxes is the cost of capital cost of disk and the operational cost of maintaining it. A lower cost storage solution is required for infrequently accessed data.
- These are some other technical considerations.
  - The Web interface for the current ECM online banking solution will have to be amended to support imaging.
  - Imaging services must be available as part of the existing SOA framework to allow partner banks to integrate with their existing systems.
  - The physical checks must be converted to a digital format and be stored into the imaging solution in an automated manner.

11.2 Designing the solution

The existing ECM solution is a FileNet environment built around Content Engine, Process Engine, and Application Engine. Though a service oriented architecture, partner banks use leveraged Web services for integrating bank offerings with bill payment and account management services in a comprehensive management interface delivered over the Internet. The addition of imaging capabilities requires more components and adding to the complexity of the exiting solution.

Figure 11-1 illustrates the ECM environment with the addition of the imaging components. Adjacent to the object store is a new component, Image Services. Image Services (IS) can either stand alone as a complete imaging solution or act as part of an ECM application as shown here. Content Engine has the capability to federate other FileNet content respositories using an adapter called Content Federation Services. Content Federation Services or CFS allows objects to be stored in an Image Services environment as if it were a WORM device. By setting a up a fixed content device in enterprise manager an IS environment is treated as a storage area. This allows the classes in Content Engine to intelligently route the check images into IS which is better suited than Content Engine for managing image data.

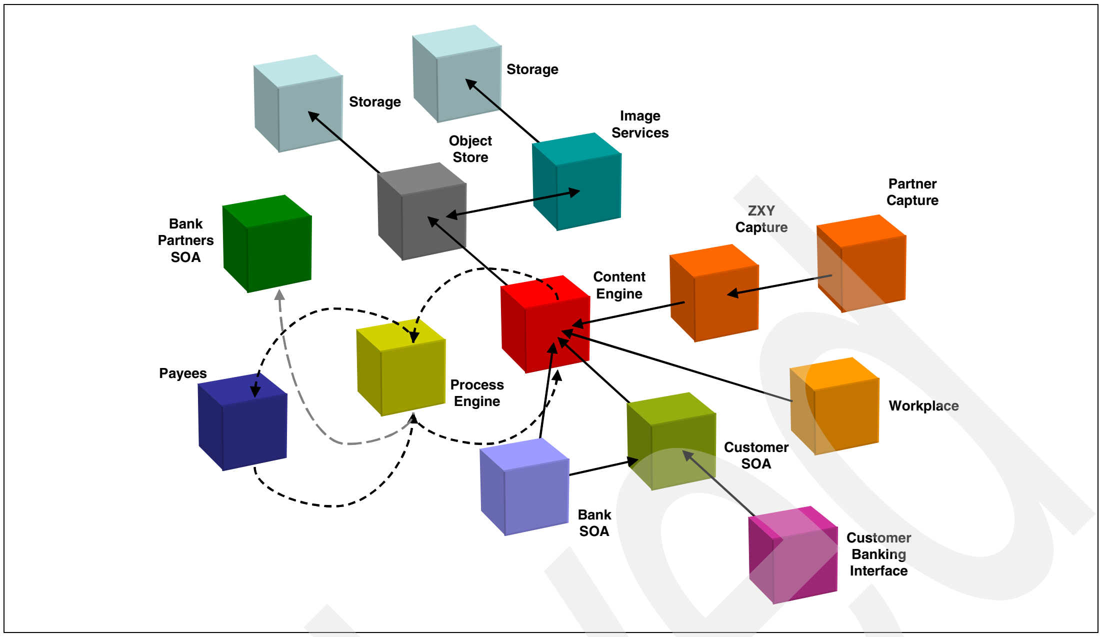


Figure 11-1 Logical ECM Design Including Document Imaging

Image Services stores content in a manner which differs significantly from Content Engine. IS was initially intended as an image archive solution allowing data to be stored to optical disk. Before the proliferation of CD and DVD media, writable optical discs were in use in enterprise environments to archive data. Many businesses required the ability to archive large quantities of data but doing so on magnetic disk was cost prohibitive. Optical discs were an inexpensive way to archive information which would not be required for frequent retrieval. These discs were stored in a library similar to a modern tape library. The disc platter was protected by a caddy with a mechanical door which exposed both sides of the disc. In Image Services terms, the writable areas of each side of a disc are known as an Optical Storage Archive and Retrieval or OSAR surface.

The logical representation of a surface was preserved in later years when Image Services began to support magnetic disk or as it is known in IS terms MSAR. An MSAR is a virtual library containing a number of surfaces, but instead of a physical disc surface, an MSAR surface is a file in which the content is written. These files can be ejected much like an optical disk and taken offline for long term storage. The benefit of this capability for ITSO will be their ability to “eject” check and fax image surfaces which are older and not likely to be required with the level of immediacy typical of an online banking application.

The drawback for MSAR from an application perspective is managing MSAR surfaces after they have been ejected. An MSAR surface file is a container with a .dat extension consisting of a header, image data, and a checksum. The files are virtually ejected removing a link to the MSAR file. Unlinked files can then be moved for archival. ITSO’s goal is to reduce the cost of storage to a point where storing the image of a check is feasible which will require storing ejected MSAR surfaces on tape to free up disk space. If a user requests a file on an ejected MSAR surface, the process of retrieving this file will involve manual intervention and maintaining a staff to manage this process could likely eliminate the cost advantage of tape. Image Services can instead be configured for Single Document Storage for IBM DR550. Doing this however is only appropriate in specific circumstances.

IBM DR550 is a protected storage platform where data tiered by automatically migrating files from high speed storage platforms to high capacity storage platforms based on age. Single Document Storage or SDS allows individual images to be stored as single autonomous entities using WORM devices such as DR550. Using SDS, Image Services no longer maintains surfaces to encapsulate data. Instead a single document is passed to the SDS API which class the appropriate DR550 API. The document is indexed in a database and is accessible from IS the same way it is from any other storage technology. The IBM DR550 storage platform will move files based on an age policy. When an image meets the age indicated by the policy, it is moved to the next designated storage pool. Storage pools are configured using fiber channel disk, for optimum speed, SATA disk for higher capacity, but slower nearline storage and tape for long term archival. The use of DR550 configured in this way is dependent on the nature of the data.

Figure 11-2 is a decay model based on the prospective number of requests ITSO expects to receive on a given day. The age of the image in months is indicated on the X axis and the request frequency is on the Y axis. As an image ages, the likelihood of being requested becomes increasingly diminutive. Each curve inclines slightly in the second month, and declines steadily until month eight where there are no images are requested. This model allows us to predict that images after nine months will likely never be viewed. This is further evidenced by our understanding of personal banking where it is common for an account holder to balance their check book at the end of the month. Once the months expenditures are balanced, it is unlikely the account holder will revisit those transactions and thus have to see images of the checks.

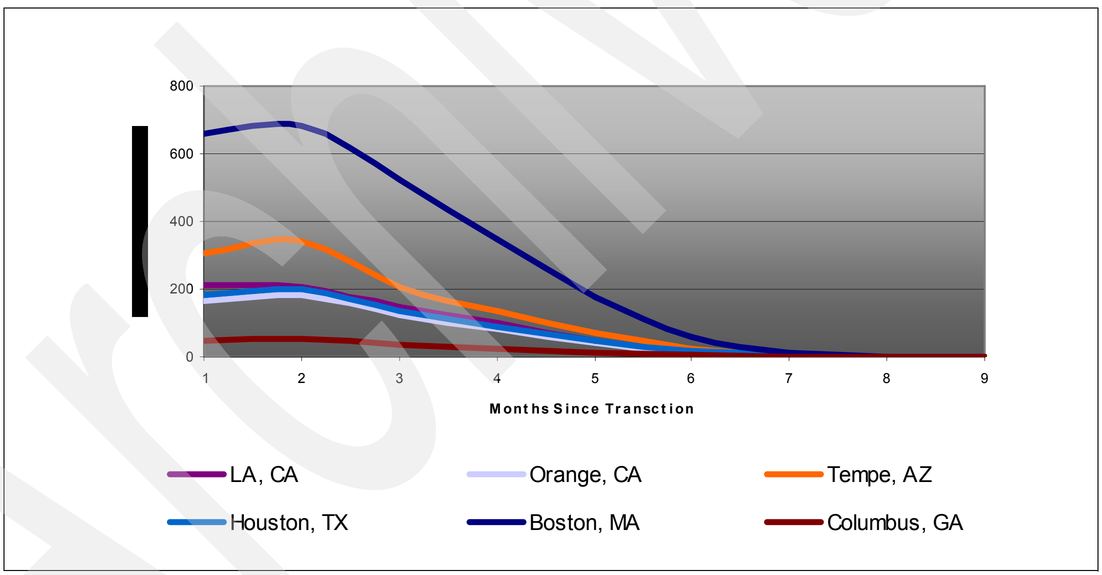


Figure 11-2 Image Request Decay Model

A check image which does not get requested is of no value to ITSO with the exception of complying of retention requirements. The use of SDS and DR550 allows images older than nine months to be moved to tape. When an image archived to tape is requested, the DR550 retrieves and returns the image stream to IS which over CFS transmits the stream to Content Engine and eventually to the users interface. The image will be cached in Content Engine so that subsequent requests for the image are handled from disk.

Figure 11-3 illustrates how images are scanned, injested by the solution, and are retrieved from the DR550. Checks are scanned at the partner banks site using IBM FileNet Capture. Capture is designed to segregate task in what is known as a capture path. Paths can be split between multiple capture stations as shown. Each partner bank scans the checks using Capture which sends the batch to the capture station at ITSO. There the image will be scanned using zonal OCR to determine the account number and the check, using a Workflow routes the check to the appropriate image folder for the user.

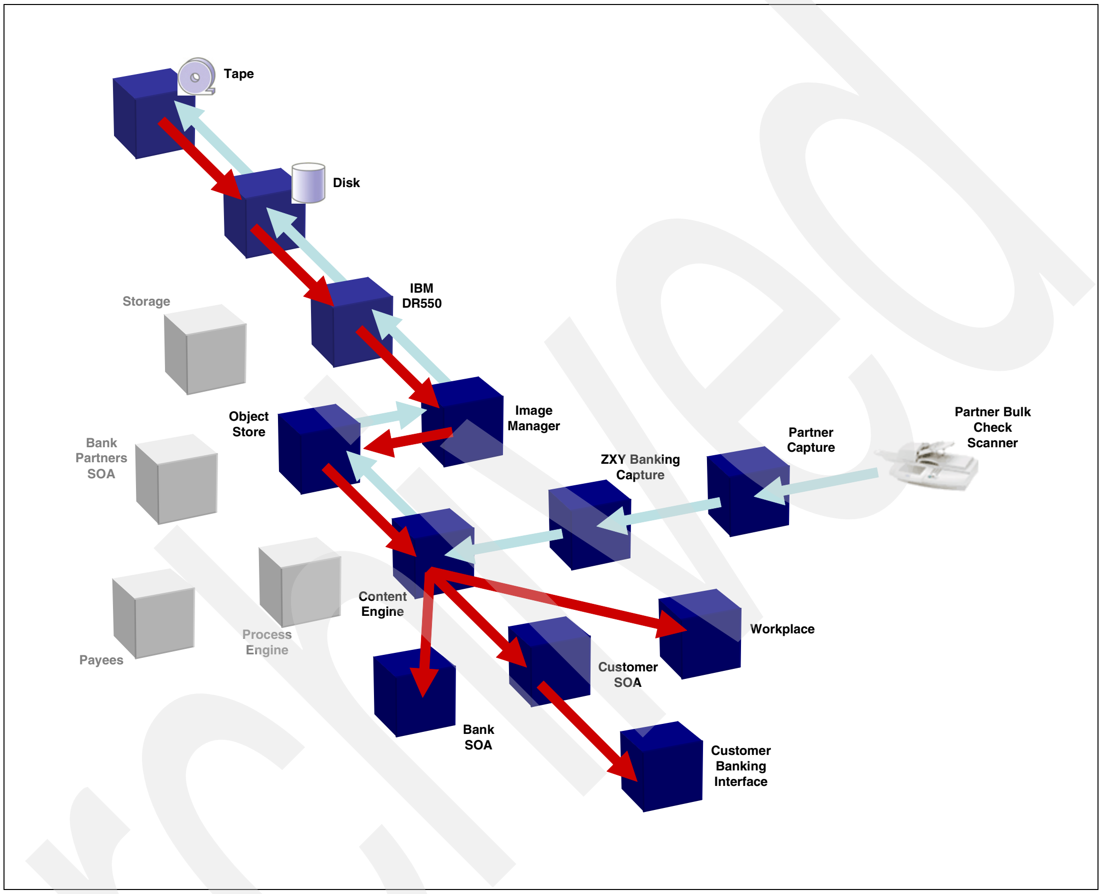


Figure 11-3 Ingestion and retrieval of check imaging using SDS and DR550

The following technical requirements will be expanded from 11.1.2, "Determining technical requirements":

- The potential cost of storing check images over seven years will likely approach hundreds of petabytes. The cost of storage will play a key role in determining the viability of providing this service.
  - IBM DR550 using SDS, CFS, and Image Services will allow images which are older than nine months and therefore are only required for retention purposes, to be stored on tape at a fraction of the cost of disk.

- The addition of check imaging must not prolong the implementation date for a new customer. Hardware additions must occur without impacting customers.
  - The Image Services addition will be scaled in the same manner as the P8 Platform solution. Infrastructure will be initially sized 50% larger than the maximum projected load required and new systems are added before capacity reaches 75%. This allows for a portion of the environment to be offline for maintenance and to accommodate organic growth. Customer data will be logically segregated based on the bank partner they do business with. The servers in the ECM solution will be running on the same platform and the operating system and software for these servers will be run from SAN disk to allow any one server to assume the role of another.
- Delivery of image data must integrate with the existing infrastructure and programatic delivery of the existing online banking system.
  - Image Services using CFS integrates seamlessly with the existing infrastructure. The class based structure of Content Engine allows the addition to be obfuscated and completed with minimal code changes.
- The primary expense of storing faxes is the cost of capital cost of disk and the operational cost of maintaining it. A lower cost storage solution is required for infrequently accessed data.
  - Infrequently accessed data will be stored on DR550 where older files will be migrated seamlessly to tape storage. Tape can be procured at a fraction of the total cost of ownership of disk storage with minimal impact to the viability of the solution.
- Imaging services must be available as part of the existing SOA framework to allow partner banks to integrate with their existing systems.
  - Content Federation Services allow the relevant image Services to be available without introducing code changes.
- The physical checks must be converted to a digital format and be injested into the imaging solution in an automated manner.
  - IBM FileNet Capture will inject the documents into the solution over the network.

11.2.1 Projecting volume growth and use patterns

ITSO Banking Corporation plans to launch the check imaging service to all six banks servicing 3.34 million total possible users. Of this amount, only a portion of the account holders will be actively using the checking feature. At launch the service will handle any bank draft such as checks and money orders scanned at the partners location.

Figure 11-4 illustrates the data projecting image view activity for this next year. It is assumed that the number of image views will be determined by a factor of the number of transactions multiplied by the percentage of users who ordered checks from their bank. Although this type of data is primarily used to size application tier components it also gives an indication of how the storage will be consumed and what times of day the storage will be busiest.

Los Angeles, CA	Orange, CA	Tempe, AZ	Houston, TX	Boston, MA	Columbus, GA	Total Checks / Day	Total Check Storage / Day in MB	Total Check Storage / Year in GB	Total Fax Storage / Year in GB
0	0	0	0	1	0	1	0	0	1
0	0	0	0	4	0	4	2	1	4
0	0	0	1	18	0	19	10	3	20
0	0	1	4	90	1	96	5	2	102
1	0	5	19	449	5	479	240	85	512
3	0	24	38	4,494	27	4,586	2,293	817	4,904
16	2	119	76	5,913	134	6,260	3,130	1,116	6,694
111	12	597	228	5,256	268	6,472	3,236	1,153	6,920
1,335	60	2,986	911	5,190	482	10,964	5,482	1,954	11,724
1,878	299	3,047	1,639	4,599	803	12,264	6,132	2,186	13,115
2,504	1,493	2,956	2,732	2,851	571	13,106	6,553	2,336	14,015
2,420	1,658	2,365	1,942	1,768	82	10,235	5,118	1,824	10,945
918	1,526	1,655	850	1,096	16	6,061	3,031	1,080	6,482
835	1,221	1,076	486	680	5	4,302	2,151	767	4,600
501	1,161	430	81	421	2	2,596	1,298	463	2,776
100	580	86	16	84	1	868	434	155	928
20	116	17	3	17	0	174	87	31	186
4	23	3	1	6	0	37	18	7	40
1	5	1	0	2	0	8	4	1	9
0	1	0	0	1	0	2	1	0	2
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
10,646	**8,157**	**15,370**	**9,025**	**32,939**	**2,398**	**78,534**	**39,224**	**13,981**	**83,979**

Figure 11-4 Image Volumes and Second Year Fax Volumes

The decay model in Figure 11-2 is based on the prospective number of image requests ITSO expects to receive on a given day. The use pattern for a given check, if it is requested will follow this model.

11.3 Overall solution architecture

ITSO Banking Corporation thus far has solidified the business requirements for the delivery of the images solution, developed its functional requirements and completed a study of data volumes and use patterns. Technical requirements have been made to reflect the business requirements, functional requirements and expected volumes. The remaining is the task of developing the infrastructure design and ensure it meets or exceeds the capabilities required of the business model. The design is broken down in terms of software and hardware.

11.3.1 Software

The additional applications used by ITSO are described as follows:

IBM FileNet Image Services 4.1

Image Services will provide a robust and scalable platform for storing check images and provide access to the IBM DR550. Image Services will be federated by Content Federation Services, a capability provided by Content Engine.

IBM FileNet Capture Professional 5.1

Capture will still be responsible for processing inbound faxes and will also process the OCR and committal processes of capture paths from partner banks. When checks are loaded, the images are scanned at the partner location using a company provided batch scanner. When the batch arrives at ITSO, using zonal optical character recognition all of the fields will be translated to metadata, the content stored, and a subsequent workflow is launched to route the image to the appropriate account. When a check can not be properly scanned it is routed to a ITSO analyst who to fill out the form data and launch the appropriate workflow.

IBM DB2 Universal Database 9.1

The Image Services application components will use DB2 as its database. DB2 will run as an instance on the current hardware dedicated to the ECM solution in place.

11.3.2 Hardware

The additional hardware required to run the application is described as follows:

Server Infrastructure

Figure 11-5 illustrates how the servers were matched to the performance characteristics outlined in section 11.2.1, “Projecting volume growth and use patterns”. The majority of the servers in the imaging solution will be running on IBM 560Q servers. The operating system and software for these servers will be run from SAN disk to allow any one server to assume the role of another. The data base which supports Image Services will be a DB2 instance running on the existing P570 infrastructure. To accommodate the growth experienced in the last year, additional servers have been added to run Content Engine, Process Engine, and Application Engine.

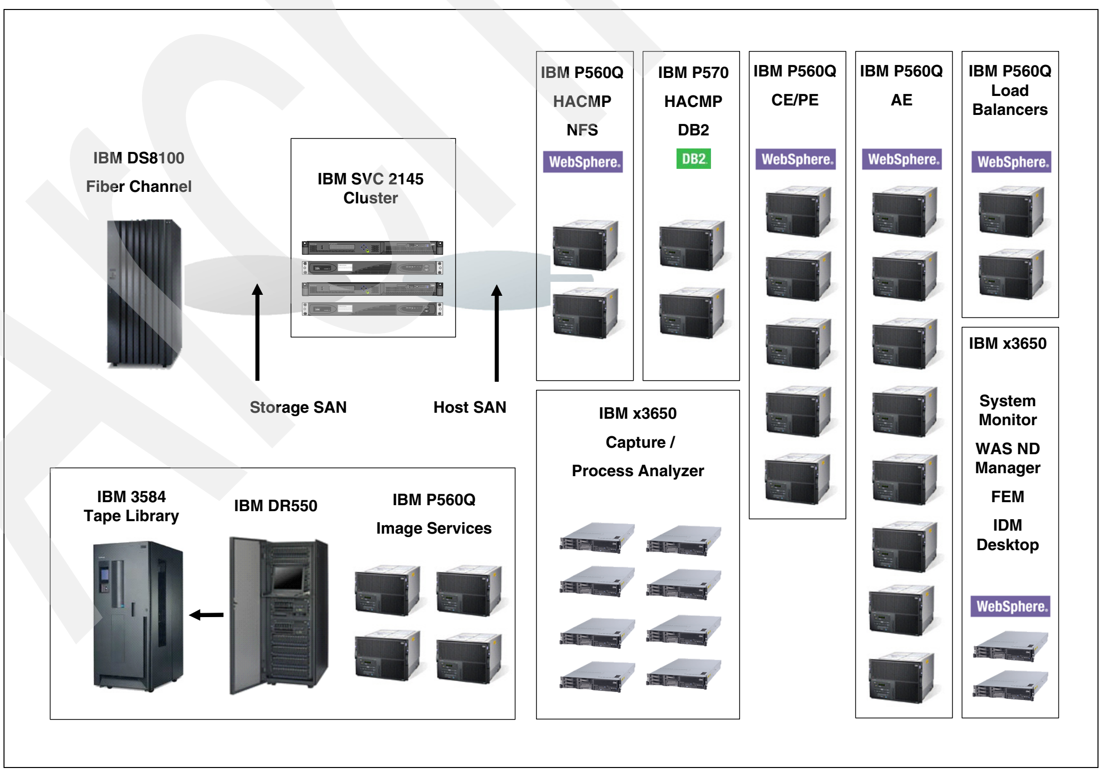


Figure 11-5 Infrastructure Changes to Support Image Services

Storage infrastructure

The IBM DR550 offers chronological and event-based data retention for check images using policies. Event-based retention enables management of images to have explicit criteria to be met before a record is deleted. Chronological retention protects records from deletion until a the retention period expires. Images will likely outlive the media standard, as such the DR550 can move images from disk to tape and from generation to generation while maintaining data immutability. This capability is achieved using a tiered storage management feature enabling image management on multiple tiers of storage (SATA disk, then tape) to provide a more cost-effective solution over SATA alone. The movement from SATA to tape will occur after an image is nine months old, when it is no longer likely to be viewed. As images reach the seven year mark, they are expunged form the system liberating resources for new images.

IBM DR550 provides a non-erasable, non-rewriteable platform and meets many regulatory standards for WORM storage. Images which require retention beyond the typical period allow a designated retention date to be suspended through a feature called a deletion-hold. This feature will enable ITSO to freeze images in the event they are required for litigation of fraud investigation reasons. Images required from tape are cached in the pre-existing Content Engine cache area st0red on fibre channel disk. This allows cached images to be quickly referenced once they are retrieved.

The added cost benefit and use model for FAX images also allows ITSO to migrate data to that platform and enjoy further cost savings. The migration from SATA to tape will be achieved by changing the storage policies for fax classes in Content Engine. This change can occur without affecting the current application or introducing an observable change to online banking customers. The costs freed by moving this data to the DR550 can be applied to the implementation of the imaging solution, further benefitting ITSO.

11.3.3 Solution summary

ITSO Banking Corporation, using this solution, can add bank draft images as a service which enables its partners to better position their online banking products, and compete with large banks without having to make an up front investment in infrastructure. To hedge the cost of ITSO’s own operational and capital expenses, this solution allows ITSO to deliver image services at a significant cost advantage to it’s partners. The use of IBM DR550 allows ITSO to be flexible in implementation and make better use of storage space than they previously could. The added capability created by Images Services and DR550 will allow the ITSO sales team to offer a stronger product while still maintaining the bottom line.

The IBM Information Infrastructure enhancement met ITSO’s business requirements as outlined:

- New infrastructure costs be kept in line with revenue. Infrastructure costs which outpace the projected revenue growth are unacceptable.
  - The IBM DR550 reduces sort and long term storage costs keeping recurring storage expenses below SATA disk.
- Infrastructure changes must not threaten the capability to start service in ten business days of the close of a new contract nor impact bank partners in a negative way.
  - Following the implementation model in use by the rest of the ECM solution, ITSO can quickly expand to meet requirements without introducing complications.

- Partner banks must be able to offer their customers the ability to display checks, money orders or any other type of paper bank order online.
  - IBM Filenet Image Services provides the capability to intelligently manage images, store them effectively and integrate seamlessly with current ECM environment. Content Engine Content Federation Services allows Image Services to be utilized as a Fixed Content Device eliminating the requirement to re-architect the software or the infrastructure.
- While the capability to fax in bank transactions has been successful, the cost of archiving this data relative to it value is prohibitive. ITSO requires a solution which allows this data to be archived for up to seven years at a lower cost.

IBM DR550 provides the full life cycle management of the image data without requiring user intervention. This capability abates litigation expose and reduce the capital costs of storage and the recurring costs to power and maintain the storage. Fax data can be seamless migrated to the platform without negatively impacting the application.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

IBM Redbooks

For information about ordering these publications, see “How to get Redbooks” on page 222. Note that some of the documents referenced here might be available in softcopy only.

- *Content Manager OnDemand Guide*, SG24-6915
- *IBM System Storage SAN Volume Controller,* SG24-6423
- *Best Practices for Setting Up an IBM CommonStore Solution For Mailbox Management, e-Mail Retention and Discovery*, SG24-7325
- *TotalStorage Productivity Center V3.3 Update Guide*, SG24-7490
- *IBM System Storage Productivity Center Deployment Guide*, SG24-7560

Online resources

These Web sites are also relevant as further information sources:

- IBM DS3000 Disk Storage Series:

  http://www-03.ibm.com/systems/storage/disk/entry/index.html
- IBM DS4000 Disk Storage Series:

  http://www-03.ibm.com/systems/storage/disk/midrange/index.html
- IBM DS8000 Disk Storage Series:

  http://www-03.ibm.com/systems/storage/disk/enterprise/index.html
- IBM Tape Drives:

  http://www-03.ibm.com/systems/storage/tape/index.html
- TS7520 tape virtualization:

  http://w3.itso.ibm.com/abstracts/sg247520.html?Open
- IBM N Series:

  http://w3.itso.ibm.com/abstracts/sg247129.html?Open
- IBM System Storage SAN Volume Controller:

  http://www-03.ibm.com/systems/storage/software/virtualization/svc/index.html
- Tivoli Storage Manager:

  http://www-306.ibm.com/software/tivoli/products/storage-mgr/
- System Storage Archive Manager:

  http://www-306.ibm.com/software/tivoli/products/storage-mgr-data-reten/index.html

- DR550:

  http://www-03.ibm.com/systems/storage/disk/dr/index.html
- GPFS:

  http://www-03.ibm.com/systems/clusters/software/gpfs/index.html
- System Storage Productivity Center:

  http://www-03.ibm.com/systems/storage/software/sspc/index.html

How to get Redbooks

You can search for, view, or download Redbooks, Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy Redbooks, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

IBM Redbooks

IBM Enterprise Content Management and System Storage Solutions: Working Together

(0.2"spine)
0.17"<->0.473"
90<->249 pages

IBM Enterprise Content Management and System Storage Solutions: Working Together

Business drivers for ECM

Matches storage products to ECM requirements

Provides practical case studies

An Enterprise Content Management (ECM) system is designed to contain unstructured information such as files, images, and drawings. Its purpose is the delivery of the right content to the right person at the right time, and in the right context.

In enterprise content management, the term *content* is used to refer to unstructured information; structured information such as database content is referred to as *data*. Although *data* is also present in ECM systems, it is used in a supportive role to help locate and manage the *content*.

All relevant information is stored either as data, content, or a combination of data and content. The data component is typically a database, and its contents are the metadata or indexes on the content. The core functionality is provided by one of more applications, which are accessible to clients or other applications. Directory servers can typically be leveraged for authorization and authentication services. The content can be stored directly on the file system, or via a storage management layer.

This IBM Redbooks publication will provide the necessary information to IBMers, business partners, and customers on how to implement FileNet ECM with IBM Storage Solutions.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-7558-00 ISBN 0738431605